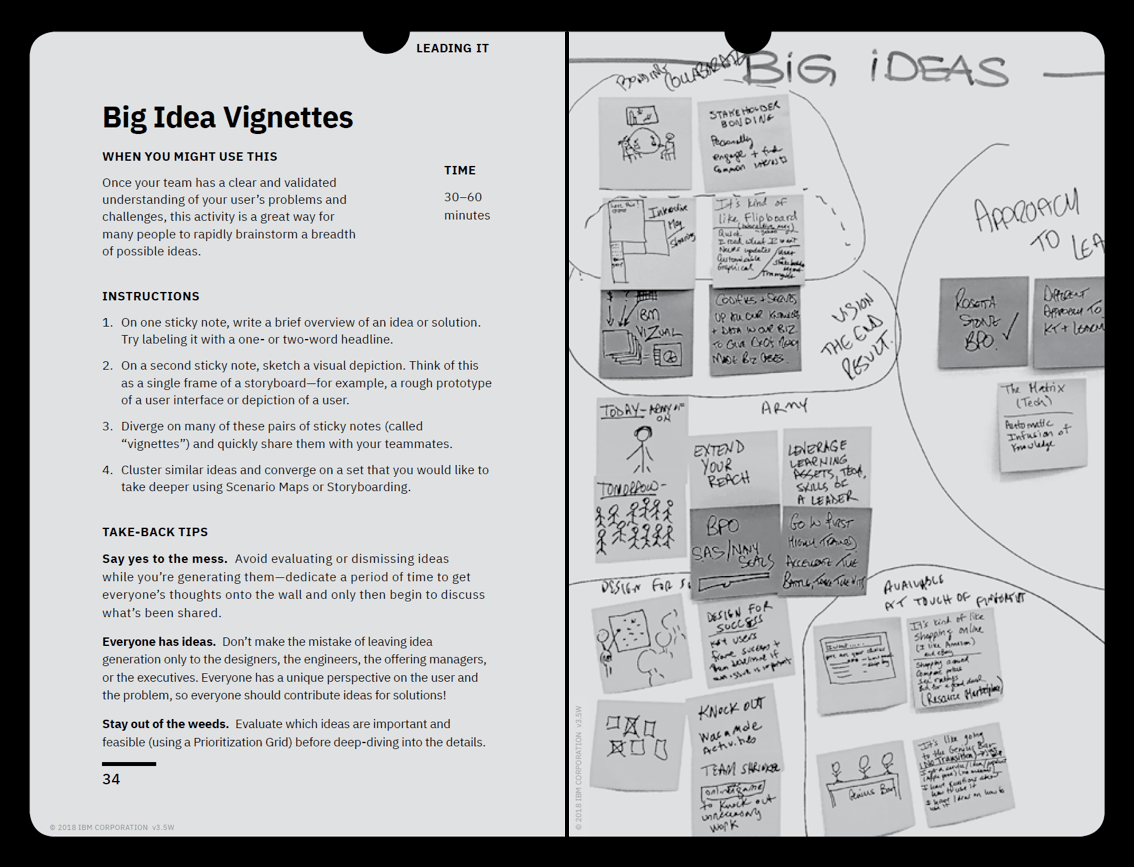
[Enterprise Design Thinking](https://www.ibm.com/design/thinking/) is our approach to applying design thinking at the speed and scale that the modern enterprise demands. It’s a framework for teaming and action. It helps our teams not only form intent, but deliver outcomes—outcomes that advance the state of the art and improve the lives of the people they serve.

The field guide provides a high-level overview of Enterprise Design Thinking:

* **LEARNING IT:** A summary of the fundamental concepts of Enterprise Design Thinking.
* **LEADING IT:** A quick reference for facilitating essential Enterprise Design Thinking activities on your team.

Are you ready to see problems and solutions from a new point of view? Embrace the principles of Enterprise Design Thinking:

* **Focus on user outcomes:** Our users rely on our solutions to get their jobs done every day. Success isn’t measured by the features and functions we ship—it’s measured by how well we fulfill our users’ needs.
* **Diverse empowered teams:** Diverse teams generate more ideas than homogeneous ones, increasing your chance of a breakthrough. Empower them with the expertise and authority to turn those ideas into outcomes.
* **Restless reinvention:** Everything is a prototype. Everything—even in-market solutions. When you think of everything as just another iteration, you're empowered to bring new thinking to even the oldest problems.



## Prerequisites

To follow along with this tutorial, you must have an [IBM Cloud account](https://cloud.ibm.com/login?cm_sp=ibmdev-_-developer-_-trial) where you can provision an instance of Watson Discovery.

## Estimated time

It should take you approximately 60 minutes to complete the tutorial.

## Steps

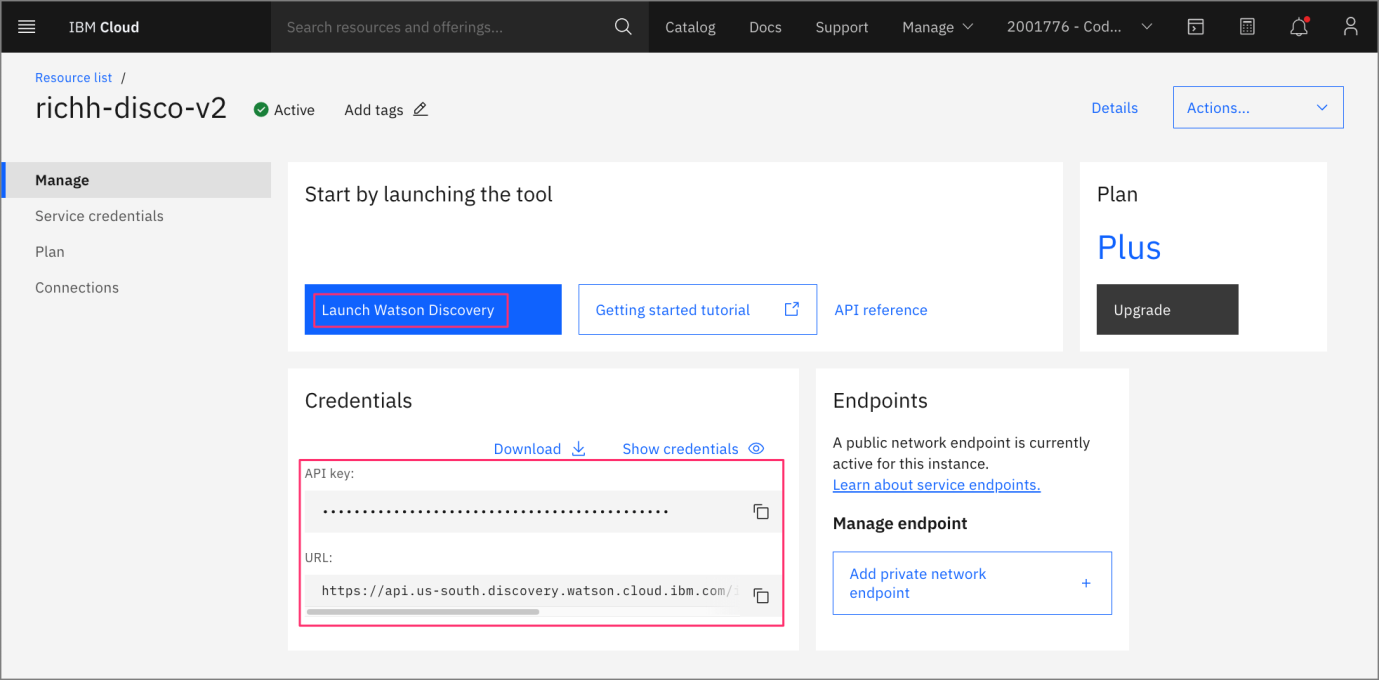
1. [Launch Watson Discovery](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#launch-watson-discovery)
2. [Create a new project](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#create-a-new-project)
3. [Upload data files into collection](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#upload-data-files-into-collection)
4. [Enrich the data](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#enrich-the-data)
5. [Add enrichments to search panel](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#add-enrichments-to-search-panel)
6. [Create your UI app](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#create-your-ui-app)
7. [Key files and concepts](https://developer.ibm.com/tutorials/create-an-intelligent-search-app-using-watson-discovery-ui-components/#key-files-and-concepts)

### Launch Watson Discovery

If you do not have an IBM Cloud account, register for a [free trial account](https://cloud.ibm.com/registration?cm_sp=ibmdev-_-developer-_-trial).

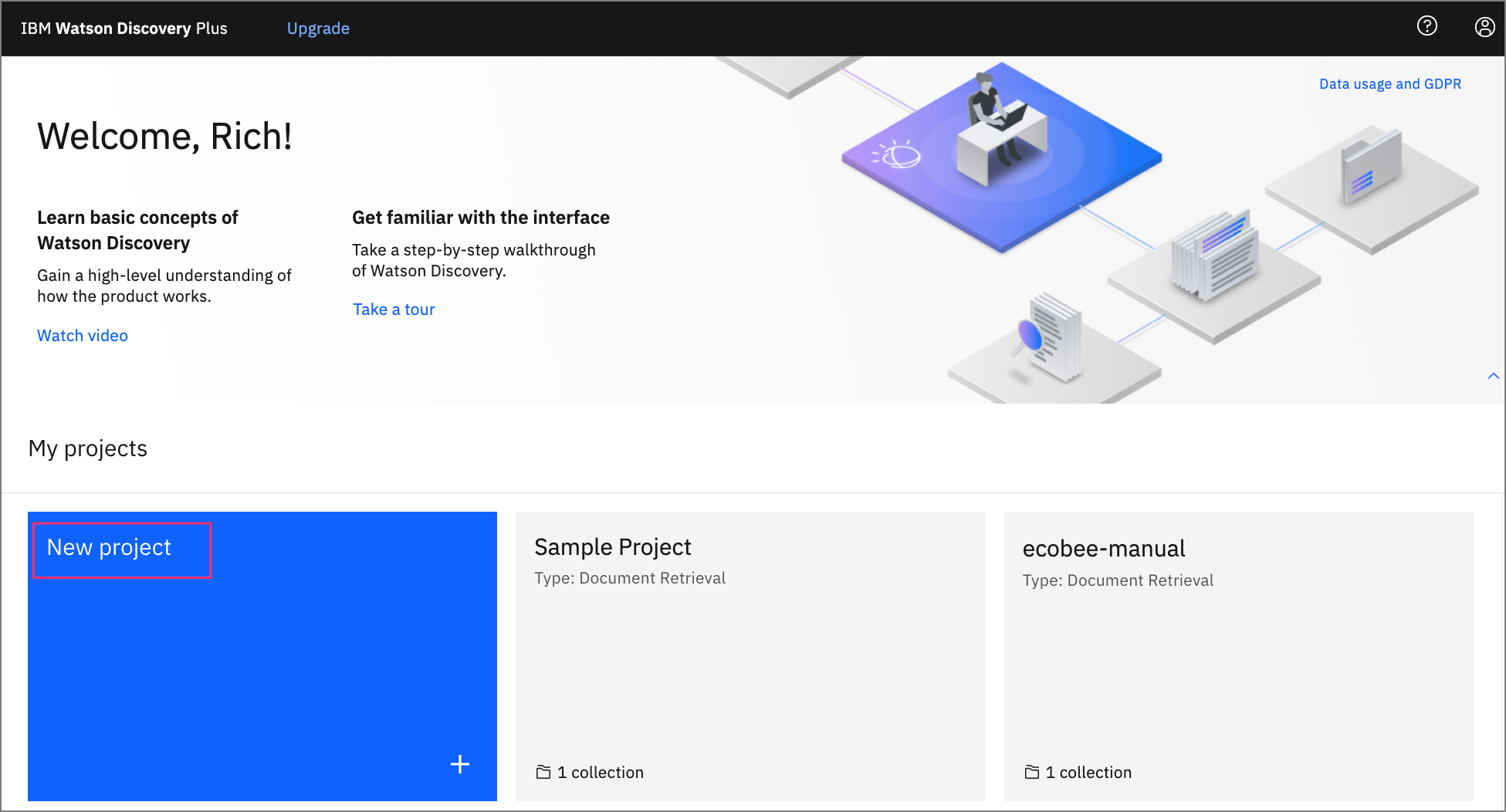
1. Using your IBM Cloud account, create a Watson Discovery instance from the [resource catalog](https://cloud.ibm.com/catalog/services/discovery?cm_sp=ibmdev-_-developer-_-trial), and select the **Plus** plan.

**Note:** The first instance of the Plus plan for IBM Watson Discovery comes with a free 30-day trial. If you no longer require your Plus instance for Watson Discovery after going through this exercise, you can delete it.



### Create a new project

1. The landing page for the Discovery service shows you a list of current projects. Click **New project**.



# onfigure the IBM Watson Assistant Chat integration for the Assistant V2 API

* **Save As PDF**[Selected topicTopic & subtopicsAll topics in contents](https://docs.servicenow.com/en-US/bundle/vancouver-servicenow-platform/page/administer/virtual-agent/task/configure-watson-assistant-topicV2.html)

Vancouver

Configure the IBM Watson Assistant Chat Integration to run a dialog skill (conversation) created in IBM Watson Assistant with the Assistant V2 API. The V2 API enables the Virtual Agent web chat client to run a topic that uses Watson Assistant's intent disambiguation feature to improve topic discovery.

**Before you begin**

With the IBM Watson Assistant v2 API chat integration, the web chat client communicates with an assistant instead of with a workspace. Workspaces are referred to as dialog skills. For information on IBM Watson Assistant, see [Migrating to the v2 API](https://cloud.ibm.com/docs/services/assistant?topic=assistant-api-migration&locale=en-us#overview) in the IBM Watson Assistant documentation.

Role required: virtual\_agent\_admin or admin

**About this task**

This integration runs an IBM Watson Assistant dialog as a Virtual Agent topic in the web chat client. It doesn't require enabling Natural Language Understanding (NLU) or setting IBM Watson Assistant NLU as the NLU service provider for Virtual Agent.

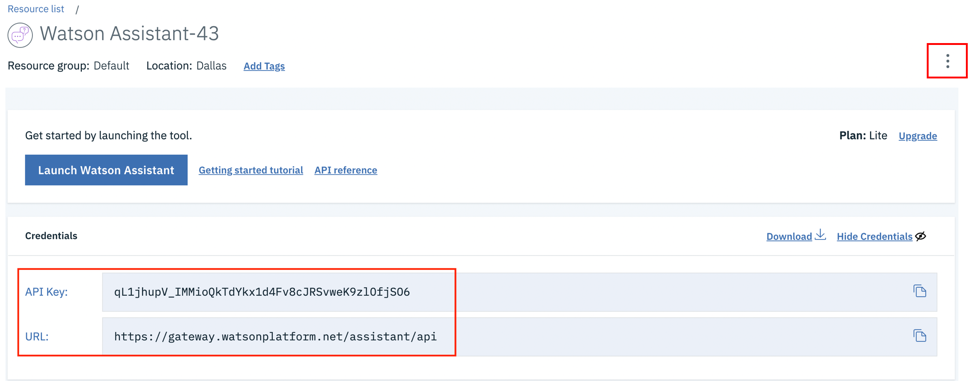
The IBM Watson Assistant V2 API enables the web chat client to use the Watson Assistant intent disambiguation feature, which can potentially reduce false-positive intent predictions. If a user's utterance is ambiguous, IBM Watson Assistant suggests additional intents that might better match the user's true intention. For example, if the user utterance is restaurant, IBM Watson Assistant can suggest other possible intents, such as book a restaurant or restaurant reviews.

**Procedure**

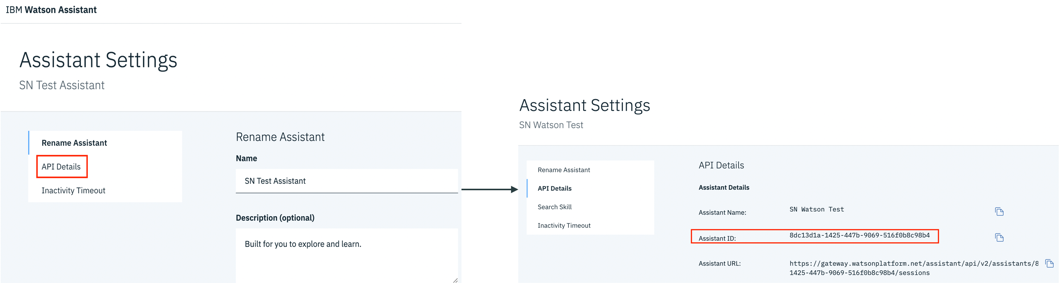
1. Configure IBM Watson Assistant.
   1. Create your assistant, then add a dialog skill.

For details, see [Creating an assistant](https://cloud.ibm.com/docs/services/assistant?topic=assistant-assistant-add) in the IBM Watson Assistant documentation.

* 1. On the Resource list page, locate and record your API key and URL, which you need to set up the chat integration:



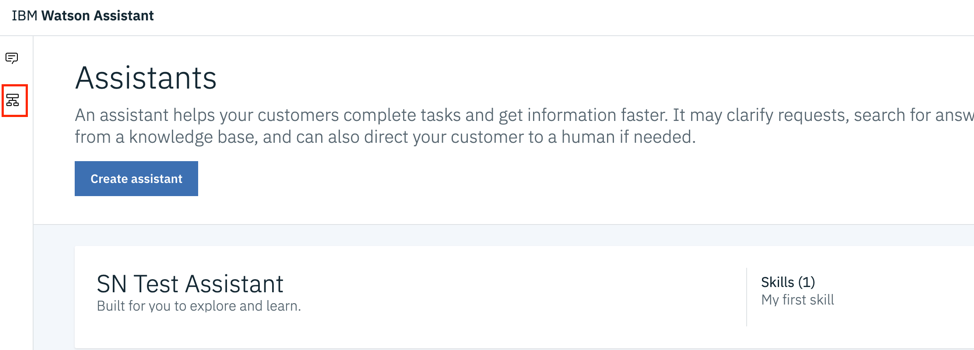
* 1. Select the three vertical dots icon https://servicenow-be-prod.servicenow.com/bundle/vancouver-servicenow-platform/page/administer/virtual-agent/images/kebab-menu.png?_LANG=enus and select **Settings**.
     + Select the **API Details** tab.
     + On the Assistant Settings page, find the **Assistant ID** key.



1. Enable the disambiguation feature in IBM Watson Assistant (for premium and plus users only).

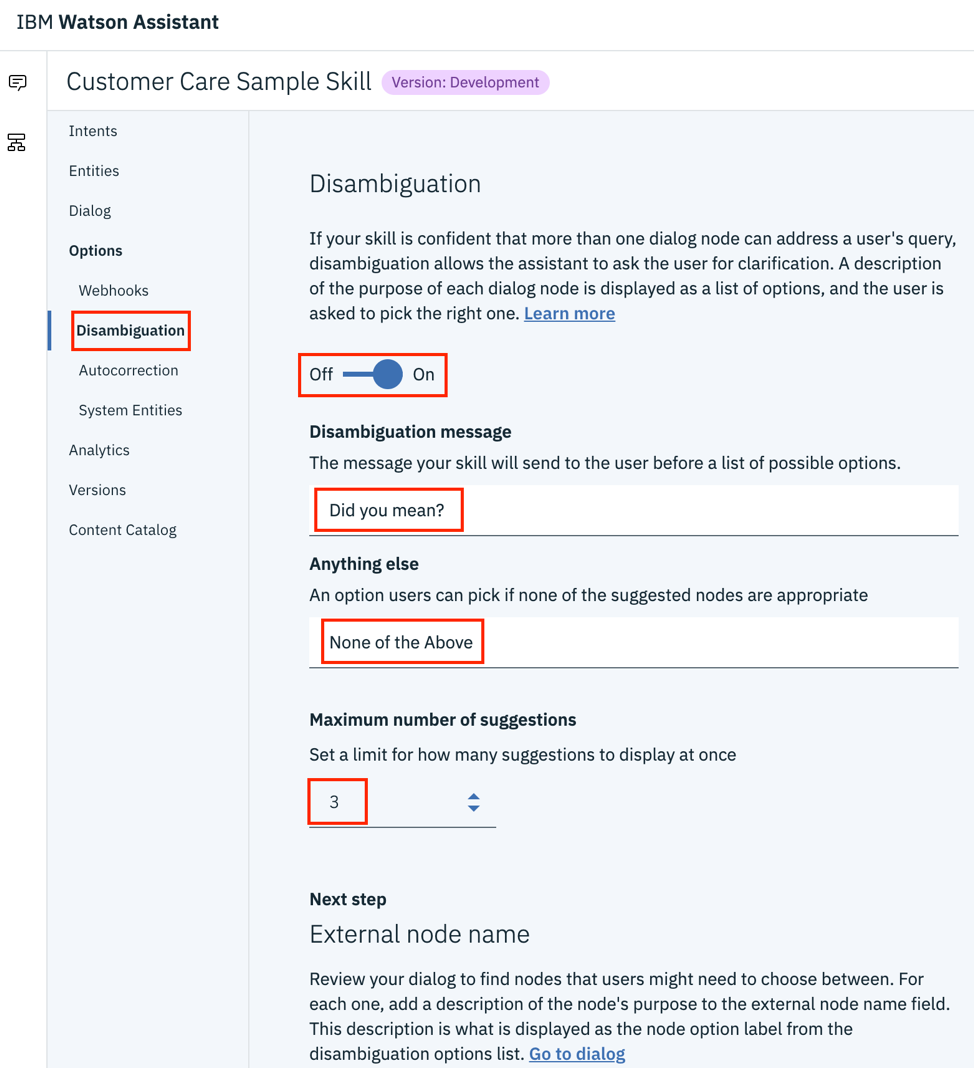
You can enable (or disable) disambiguation for each skill, and also specify a disambiguation message and the number of suggestions to be displayed to the end user.

* 1. Access your skill by selecting the skill tree icon https://servicenow-be-prod.servicenow.com/bundle/vancouver-servicenow-platform/page/administer/virtual-agent/images/IBM-watson-skilltree.png?_LANG=enus in the panel.

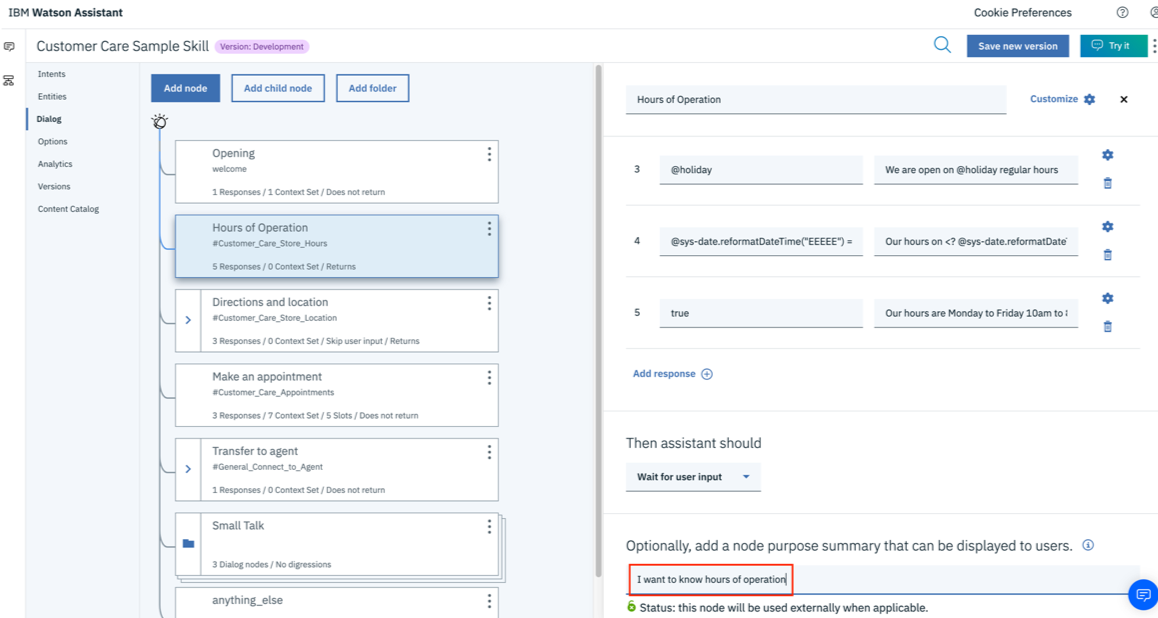


* 1. Navigate to **Options** > **Disambiguation** to enable disambiguation and set the disambiguation features.

The following example shows the disambiguation features that can be set, such as the disambiguation message and the number of suggested intents to be displayed to the end user.



1. Navigate to **Dialog**, select a node to which the disambiguation feature will apply, and provide a summary of the node that will be displayed to users if it's one of the intent suggestions.

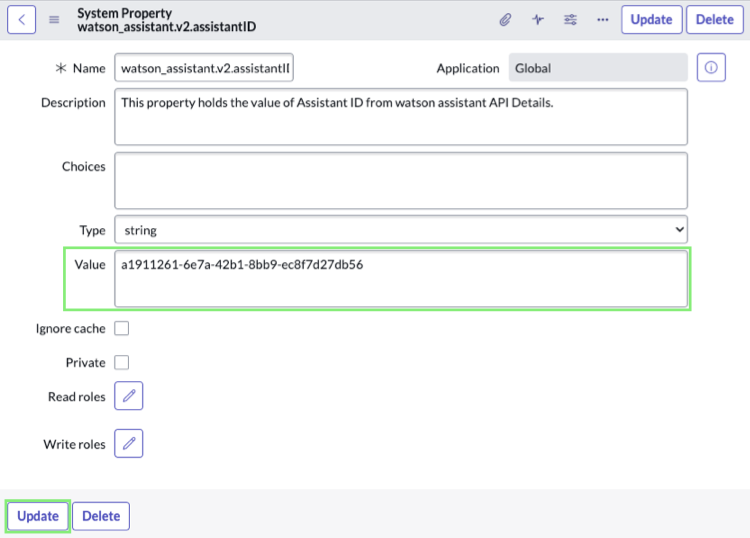


1. In your ServiceNow instance, activate these plugins:
   1. Glide Virtual Agent (com.glide.cs.chatbot), if not already activated (requires a subscription)
   2. IBM Watson Assistant Integration V2 plugin (com.glide.cs.ibm.watson.assistant.topicV2)
2. In Virtual Agent Designer, publish the IBM Watson Assistant V2 topic:
   1. Navigate to **All** > **Conversational Interfaces** > **Virtual Agent** > **Designer**.
   2. In the Topics page, select the IBM Watson Assistant V2 topic to open the Topic Properties page.

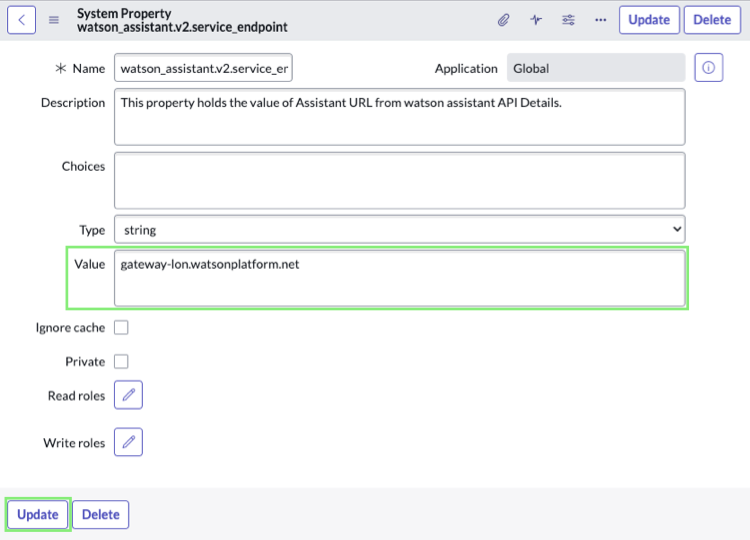
Notice that the **Keyword** for this topic is IBM Watson Assistant V2.

* 1. In the Topic Properties page, select **Publish** to deploy the topic to your Virtual Agent web client.

1. Navigate to **All**, and then enter sys\_properties.list in the filter.
2. Update the following properties.
   1. watson\_assistant.v2.assistantID — In the **Value** field, enter the Watson Assistant ID for your assistant and select **Update**.



* 1. watson\_assistant.v2.service\_endpoint — In the **Value** field, replace the existing value with your Watson Assistant URL and select **Update**.



1. **Create a Watson Assistant Instance**:
   * Set up an instance of Watson Assistant on the IBM Cloud platform or your preferred Watson Assistant provider.
2. **Obtain API Credentials**:
   * Obtain the API credentials (API key and URL) for your Watson Assistant instance. These credentials will be used to authenticate with Watson Assistant from ServiceNow.
3. **Configure Virtual Agent**:
   * Log in to your ServiceNow instance and go to Virtual Agent.
4. **Create a New Topic**:
   * In ServiceNow Virtual Agent, create a new topic or select an existing one where you want to integrate Watson Assistant.
5. **Configure the Topic**:
   * Within the topic, you can specify the questions or phrases that trigger the Watson Assistant integration. Set up the initial dialog and interaction flow.
6. **Integrate with Watson Assistant**:
   * Within the topic configuration, there should be an option to integrate with Watson Assistant. You'll need to enter the API key and URL you obtained in step 2.
7. **Define Dialog and Responses**:
   * Configure the dialog and responses that Watson Assistant should provide. You can use Watson Assistant's intents and entities to define the conversation logic.
8. **Test and Validate**:
   * Test the integration to ensure that Watson Assistant responds appropriately to user queries within the ServiceNow Virtual Agent.
9. **Handle Escalations**:
   * Configure what happens when Watson Assistant can't handle a query. You might set up escalation options, like routing the conversation to a live agent.
10. **Training and Fine-Tuning**:
    * Continuously monitor the performance of the chatbot and fine-tune its dialog and responses to improve accuracy and user satisfaction.

The specific source code you would use for the integration would be the API request code that sends user queries to Watson Assistant and processes the responses within the context of the ServiceNow Virtual Agent. This code can vary depending on the scripting language and libraries used within ServiceNow.

Typically, you would use JavaScript or other scripting languages supported by the ServiceNow platform to make API requests to Watson Assistant. Here's a simplified example of how you might structure such a request in JavaScript:

// Define API credentials

const watsonApiKey = 'YOUR\_WATSON\_API\_KEY';

const watsonUrl = 'YOUR\_WATSON\_API\_URL';

// User input

const userInput = 'Hello, chatbot!';

// Make an API request to Watson Assistant

const response = gs.ajax({

url: watsonUrl,

method: 'POST',

headers: {

'Authorization': `Bearer ${watsonApiKey}`,

'Content-Type': 'application/json'

},

data: JSON.stringify({

input: {

message\_type: 'text',

text: userInput

}

})

});

// Process the response from Watson Assistant

const watsonResponse = JSON.parse(response.getBody());

// Send watsonResponse back to the Virtual Agent for handling

// Further interaction and routing logic can be implemented here.