6.2 AI Scenario Producer

As an AI scenario producer, you develop AI use cases (scenarios) on one or more AI runtimes and productize them.

An AI runtime is the infrastructure on which your AI technology is implemented. SAP AI Core (AI runtime) is compatible with SAP AI Launchpad. The assumption throughout this guide is that SAP AI Core is used as the AI runtime.

When you are responsible for managing the lifecycle of the AI scenario, SAP AI Launchpad provides support in the following ways:

- You can manage the operations and lifecycle tasks of the AI scenarios on multiple instances of AI runtimes, such as SAP AI Core, for all of your AI scenario consumers.
- AI scenario producers can access logs to provide support to AI scenario consumers.

Based on the roles assigned to you, within SAP AI Launchpad, you can:

- Use the *Workspaces* app to create and manage connections to multiple instances of AI runtimes which implement the AI API (such as SAP AI Core. For more information, see Workspaces [page 158].
- Use the *ML Operations* app to manage tasks over the lifecycle of your AI scenarios. For more information, see ML Operations [page 196].

Related Information

What Is SAP AI Core?

6.2.1 Prerequisites

- You have a global account for SAP Business Technology Platform and an active subscription for SAP AI Launchpad. For more information, see SAP AI Core Initial Setup.
- You have a valid subscription for an AI runtime platform such as SAP AI Core.
- You have set up an instance of the AI runtime that you will use connect to SAP AI Launchpad. If you are using SAP AI Core as your AI runtime, see Initial Setup for SAP AI Core.

6.2.2 Workspaces

You use the *Workspaces* app to create and manage connections between SAP AI Launchpad and your AI runtimes (for example, SAP AI Core). The app lets you switch between your AI runtime instances so that you can carry out further actions.

AI runtimes provides resources to process AI activities such as training and inference. You can connect any AI runtime that implements the AI API to SAP AI Launchpad. For more information, see AI API.

Accessing the App

You'll find the *Workspaces* app in SAP AI Launchpad in the left navigation panel.

After selecting a connection in *Workspaces* app, you then set the resource group as a context in which to perform ML operations. You can switch between connections and the resource groups they contain at any time. For more information, see Resource Groups.
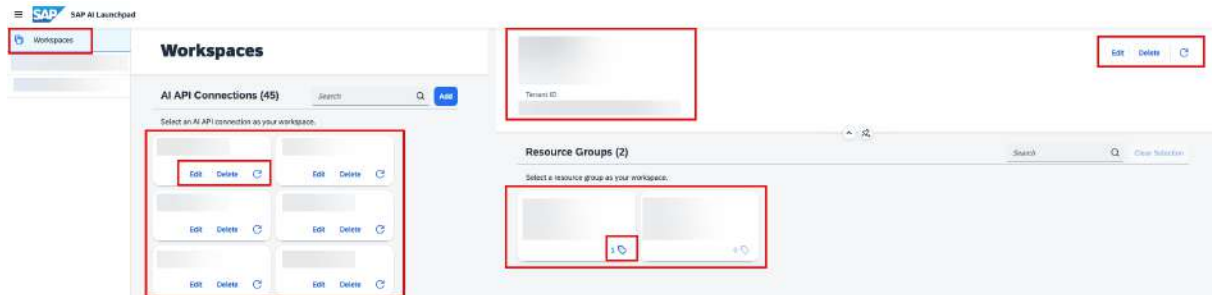
Users can edit, delete, and refresh connections. Refreshing a connection fetches the capability metadata of the underlying AI runtime (and the specific implementation of the AI API).

For more information about capabilities, their implementation, and their effect on the SAP AI Launchpad user interface, see Custom Runtime Capabilities Using the Meta API [page 43] and AI API Runtime Implementations.

> ⓘ Note
>
> In addition to the manual refresh option, SAP AI Launchpad automatically refreshes connections and capability metadata periodically.

Key Tasks

Before you can carry out any AI operations in SAP AI Core, your system administrator must complete the following tasks within the *Workspaces* app:

- Add Connection to SAP AI Core [page 160]
- Manage a Connection [page 162]
- Set Resource Group [page 164]

6.2.2.1   Add Connection to SAP AI Core

As an administrator, you can add multiple connections to different instances of SAP AI Core. You can enter the service key details for a connection manually, or upload a service key file.

Prerequisites

You have the `connections_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

Procedure

1. In SAP AI Launchpad, open the *Workspaces* app and choose *Add*. The *Create AI API Connection* dialog appears.
2. Enter a name for your new connection.

   Connection names must comply with the following criteria:
   - Contain only lowercase alphanumeric characters, hyphens (-), or periods (.)
   - Start with an alphanumeric character
   - End with an alphanumeric character
3. Upload the service key file for your SAP AI Core instance, if available.

   To upload the service key as a `.TXT` or `JSON` file, choose ⬆ *(Upload)* . Search for and choose the local service key file.

   Service key data then defaults to the remaining fields including the connection type.

4. If a service key file is not available, complete the following based on the service key and connection type you have:

   If you have a service key containing client secret credentials, choose *Secret* and complete the following:

   a. Enter the `AI_API_URL` from your service key for your SAP AI Core instance.
   b. Enter the `url` from your service key for your SAP AI Core instance.
   c. Enter the `clientid` from your service key for your SAP AI Core instance.
   d. Enter the `clientsecret` from your service key for your SAP AI Core instance.

   If you have a service key containing x509 certificate credentials, choose *Certificate* and complete the following:

   a. Enter the `AI_API_URL` from your service key for your SAP AI Core instance.
   b. Enter the `certurl` from your service key for your SAP AI Core instance.
   c. Enter the `clientid` from your service key for your SAP AI Core instance.
   d. Enter the `certificate` from your service key for your SAP AI Core instance.
   e. Enter the `key` from your service key for your SAP AI Core instance.

   Sensitive fields are masked. You can unmask to show your entry if needed.

Results

The new connection appears in the *Workspaces* app.

6.2.2.2 Manage a Connection

As an administrator, you can delete, update, or refresh a connection using the *Workspaces* app.

Prerequisites

You have the `connections_editor` role or a role collection that contains it. For more information, see Roles and Authorizations [page 321].

6.2.2.2.1 Edit a Connection

Procedure

1. In the *Workspaces* app, find the connection.

   Each connection appears as a tile. You can search the connections by entering the name or part of the name in the Q *(Search)* field.

2. Select the connection and choose *Edit* to edit the connection's details.

   The *Update AI API Connection* dialog box appears.

3. Change the connection details as required.

   > → Tip
   >
   > You can't change a connection name once it has been created, However, you can delete the connection (no data is lost) and add a new connection to the same runtime instance with a new name. See Delete a Connection [page 163].
   >
   > Sensitive fields are masked and entries are not shown in clear text. To ensure these entries are accurate, copy and paste the values from your service key.
   >
   > The *Connection Type* can be changed from clientSecret to x509 certificate and from x509 to clientSecret.

   • You can upload the service key file for your SAP AI Core instance, if available.
     To upload the service key as a When editing a connection, the `.TXT` or `JSON` file, choose
     ⬆ *(Upload)*When editing a connection, the . Search for and choose the local service key file.

Service key data then defaults to the remaining fields.

4. Choose *Update*.

Results

The updated connection appears in the *Workspaces* app.

To manage repositories, applications, and docker registry secrets for the connection, see Administration [page 167].

6.2.2.2.2   Delete a Connection

Procedure

1. In the *Workspaces* app, find the connection.

   Each connection appears as a tile. You can search the connections by entering the name or part of the name in the Q *(Search)* field.

2. Select the connection and choose *Delete* to remove the connection from SAP AI Launchpad.

   > ⓘ Note
   >
   > No data is lost from your runtime instance to which the connection points. You can add a connection again later.

Results

The connection no longer appears in the *Workspaces* app.

6.2.2.3 Set Resource Group

You select a connection and resource group so that you can complete tasks within the *ML Operations* app.

Prerequisites

- You have the `connections_viewer` role. For more information, see Roles and Authorizations [page 321].
- You have created one or more AI API connections (see Add Connection to SAP AI Core [page 160]).

Context

A connection links SAP AI Launchpad via the AI API to a specific instance of an AI runtime, for example, SAP AI Core.

Resource groups represent a virtual collection of related resources within the scope of your AI runtime. Resource groups are used to isolate related ML resources and workloads for logical and performance reasons. Administrators can create, edit, or delete resource groups using the *ML Operations* app.

By default, your users can access all resource groups. If required, you can define custom settings to connections and resource groups. Custom settings define user access at both connection and resource group level. Applying custom settings means that users only see connections and resource groups which are applicable to their needs.

Related Information

Custom Access for Connections [page 331]
Custom Access for Resource Groups [page 333]
Manage Resource Groups [page 187]
Resource Groups [page 44]

6.2.2.3.1 Assign Connection to Workspace

Context

A workspace provides users with connections to AI runtimes, such as SAP AI Core.

Procedure

1. In the *Workspaces* app, find the connection.

   Each connection appears as a tile. You can search the connections by entering the name or part of the name in the 🔍 *(Search)* field.

2. Select the required connection.

   The name of the selected connection appears in the SAP AI Launchpad header, and in the pane header for the resource groups. For reference, the tenant ID is displayed. You can collapse or pin the header.
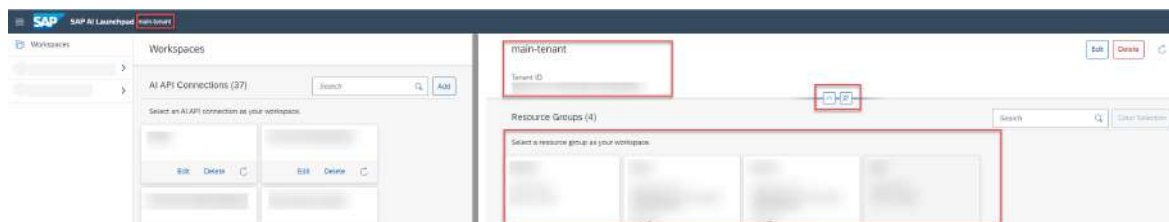
Results

With the connection assigned, you can now choose a resource group. See Assign Resource Group to Workspace [page 165].

Related Information

Custom Access for Connections [page 331]

6.2.2.3.2   Assign Resource Group to Workspace

Prerequisites

You have the `allow_all_resourcegroups` role, or you are assigned a role collection that contains it.

Context

Resource groups are used to isolate related ML resources and workloads for logical and performance reasons.

By default, users can access all resource groups. However, custom access settings may apply so that users only see connections and resource groups applicable to their needs. See Custom Access for Resource Groups [page 333].

Administrators can create, edit, or delete resource groups using the *ML Operations* app.

Procedure

1. Assign the connection to your workspace from the AI API connections list. See Assign Connection to Workspace [page 164].

   Each resource group appears as a tile in the right pane. You can search the resource groups by entering the resource ID or part of the ID in the 🔍 *(Search)* field.

   Resource groups are listed by resource ID, and with identifying details such as subaccount ID, zone ID, instance ID, and created on timestamp.

2. **Optional:** Check the labels for the resource group by choosing 🏷️ *(Labels)*. The *Labels* dialog appears showing all labels for the selected resource group. Labels include subaccount ID, zone ID, instance ID, as well as any other defined labels. Labels are subject to character limitations, and you can mouseover a label for its complete details.

3. Find the resource group and select it for your workspace. The ID of the selected resource group appears next to the connection name in the header.

   > → Tip
   >
   > If you can't find the required resource group, contact your administrator.

Results

The *ML Operations* app and its functions are now available in the navigation pane.

Related Information

Create a Resource Group [page 188]
Resource Groups [page 44]
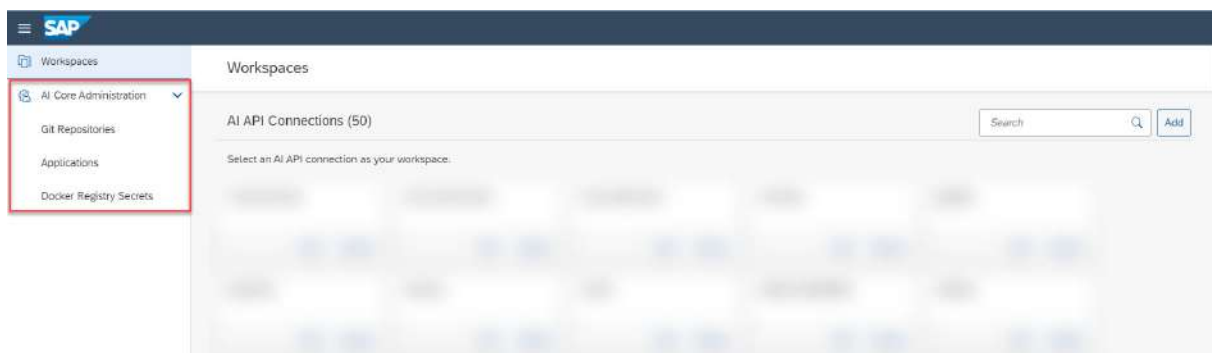
6.2.3 Administration

You use the *SAP AI Core Administration* app in SAP AI Launchpad to manage administration activities for your SAP AI Core runtime.

For example, you can use the *SAP AI Core Administration* app to manage authentications required when processing AI workflows.

Accessing the App

The *SAP AI Core Administration* app is available in SAP AI Launchpad in the left navigation panel.

After you've selected a connection in *Workspaces* app, the *SAP AI Core Administration* app is active. You can then manage Git repositories, applications, and Docker registry secrets that are connected to your underlying SAP AI Core runtime.

Key Tasks

The key tasks for the *SAP AI Core Administration* app include:

- Manage Repositories [page 168]
- Manage Applications [page 177]
- Manage Docker Registry Secrets [page 182]

Related Information

Setting Up Your Git Repository

6.2.3.1 Manage Repositories

You can maintain a list of Git repositories for use within your AI processes.

Context

Git repositories are used for storing versions of training and serving templates.

You use the *SAP AI Core Administration* app to add Git repositories to your instance of SAP AI Core (your AI runtime). You can add multiple repositories, edit their details, or remove a repository if it is no longer required. The Git repositories you define are available for selection and use in applications.

Related Information

[Setting Up Your Git Repository](#)

6.2.3.1.1 Add a Git Repository

As a system administrator, you can add Git repositories which can be used within your training and serving processes.

Prerequisites

- You've completed the initial setup. For more information, see Initial Setup.
- You have access to a git repository over the Internet.
- You've generated a personal access token for your git repository. For more information, see Create a Personal Access Token ↗ .
- If you want to onboard a git repository hosted on GitLab, make sure that the repository URL contains the .git suffix.
- Secrets aren't permitted in your repository. If secrets are used, it isn't possible to synchronize content.

You have the `aicore_admin_repositories_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

> ⓘ Note
>
> When you synchronize resources, make sure that there are no naming collisions, especially if you use multiple repositories or applications in one tenant. If you experience difficulties during synchronization, we recommend that you use only one repository or application per tenant.
>
> For example, the following repository URLs are all considered the same repository:
>
> - `https://github.com/user/repo`
> - `https://github.com/user/repo/`
> - `https://github.com/user/REPO/`

Context

Git repositories are managed by creating personal access tokens and registering them in SAP AI Core. Personal access tokens are a means of allowing and controlling connections to GitHub repositories without compromising your credentials.

You can add multiple Git repositories for a selected connection. The Git repository must already exist with valid authentication details.

> ⓘ Note
>
> SAP AI Core supports the use of Git repositories; the use of other private or open-source repositories isn't supported.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Git Repositories*.

   The *Git Repositories* screen appears with details of any existing repositories, including name, status, and URL.
3. Choose *Add* to enter reference details for a new repository.
4. Complete the fields in the *Add a Git Repository* dialog box as follows:
   a. Enter a URL for the repository.
   b. Enter a name for the repository.

      Repository names must comply with the following criteria:

      - Contain only lowercase alphanumeric characters, hyphens (-), or periods (.)
      - Start with an alphanumeric character
      - End with an alphanumeric character

      c. Enter a user name and access token.

          The combination of user name and access token enables read access to the repo.

  5. Choose *Add* to add the repo to the available repositories for the selected SAP AI Core connection.

Results

The new repo appears on the *Git Repositories* screen. The onboarding status for the new repo is *In progress* until the authorization details are authenticated.

Related Information

Add Connection to SAP AI Core [page 160]
Setting Up Your Git Repository

6.2.3.1.2   Edit a Git Repository

As a system administrator, you can edit Git repositories which are used within your training and serving processes.

Prerequisites

You have the `aicore_admin_repositories_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

You have access to the Git repository over the Internet.

Context

You can change details for a Git repository if, for example, the name you've defined isn't unique or its authentication details have changed.

> ⓘ Note
>
> You can't change the URL for the Git repository. If the URL is no longer valid or contains errors, you need to remove the Git repository and re-create it with the correct reference details. See Remove a Git Repository.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Git Repositories*.

   The *Git Repositories* screen appears with details of repositories for the connection, including name, status, and URL.

3. Find the tile for the repo and choose *Edit*.
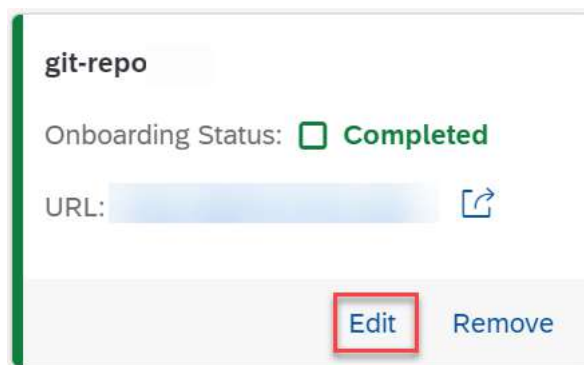
   The *Edit Git Repository* dialog box appears.

4. Change the reference details in the *Edit Git Repository* dialog box as required:
   a. Edit the name for the repository.

      Repository names must comply with the following criteria:

      - Contain only lowercase alphanumeric characters, hyphens (-), or periods (.)
      - Start with an alphanumeric character
      - End with an alphanumeric character
   b. Edit a user name and access token.

      The combination of user name and access token enables read access to the Git repository.

5. Choose *Edit* to update the repo details with the changes.

Results

The updated repo appears on the *Git Repositories* screen.

6.2.3.1.3   Remove a Git Repository

As a system administrator, you can remove a Git repository from use within your training and serving processes.

Prerequisites

You have the `aicore_admin_repositories_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

You have access to the Git repository over the Internet.

Context

You remove a Git repository from a connection if its URL is invalid or contains errors, or if the repo is no longer required. Once a Git repository is removed, it can no longer be selected as a source repository for an application.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Git Repositories*.

   The *Git Repositories* screen appears with details of repositories for the connection, including name, status, and URL.
3. Find the tile for the repo and choose *Remove*.
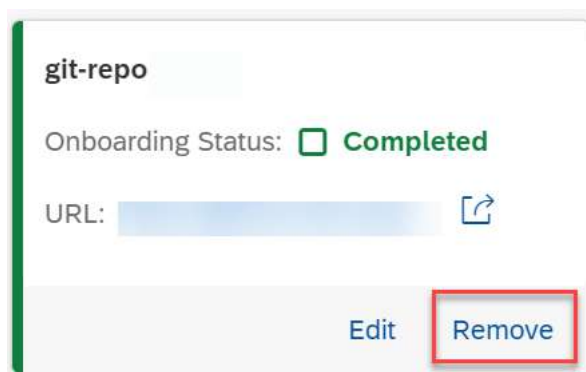
The *Remove Git Repository* dialog box appears.

4. Choose *Remove* to confirm the removal, and remove the repo from the connection.

Results

The repo no longer appears on the *Git Repositories* screen. Any applications that use the removed repo as a source repository no longer work.

6.2.3.2 Manage Object Store Secrets

You can connect your AI processes with a cloud object store, and manage access using an object store secret.

A cloud object store enables you to store files that you use in your launchpad processes. Files can include models, datasets, result sets, or other types of artifact used within model training, serving, or inferencing processes.

Cloud storage credentials are managed using secrets. Secrets are a means of allowing and controlling connections across directories and tools, without compromising your credentials.

Various hyperscaler object stores are supported, including Amazon S3, Alicloud Object Storage Service (OSS), Azure and SAP HANA Cloud, data lake (WebHDFS).

6.2.3.2.1 Add an Object Store Secret

As an administrator, you can add object store secrets for use within your AI processes.

Prerequisites

- You're using the `extended` service plan. For more information, see Service Plans [page 45].
- You have the `aicore_admin_objectstoresecret_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

Context

You can use the *SAP AI Core Administration* app to add secrets for multiple object stores. The object stores must already exist with valid credentials.

Supported cloud object stores include Amazon S3 (S3), Alibaba Cloud Object Storage Service (OSS), Azure, and SAP HANA Cloud, data lake (WebHDFS).

Procedure

1. In the *Workspaces* app, choose the AI API connection and resource group.
2. Open the *SAP AI Core Administration* app and choose *Object Store Secrets*.

   The *Object Store Secrets* screen appears with a tile for each existing secret.
3. Choose *Add* to enter reference details for a new secret.
4. Complete the fields in the *Add Object Store Secret* dialog box as follows:

   a. Confirm the resource group. To change the resource group, choose ⧉ *(Change Value)*.
   b. Enter a name for the secret.

      Secret names must comply with the following criteria:

      * Contain only lowercase alphanumeric characters, hyphens (-), or periods (.)
      * Start with an alphanumeric character
      * End with an alphanumeric character
   c. Choose the type of object store.
   d. Enter the path prefix. The path prefix is used to differentiate between different projects that are stored in the same location.
   e. Complete the information requested in the dialog box.

      > ⓘ Note
      >
      > The type of object store determines what fields are required for the object store secret.

      For S3:
      * Choose `Verify SSL` to apply the SSL security protocol to data transferred from the object store.
      * Choose `Use HTTPS` to apply the HTTPS communication protocol to data transferred from the object store.
   f. Enter the secret in JSON format. For more information, see Register Your Object Store Secret.
5. Choose *Add* to save the secret details.

Results

The new secret appears on the *Object Store Secrets* screen.

The saved secret enables read access to the nominated hyperscaler object store, enabling stored files to be used in your launchpad processes.

6.2.3.2.2 Edit an Object Store Secret

As an administrator, you can edit object store secrets used within your AI processes.

Prerequisites

You have the `aicore_admin_objectstoresecret_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

You have access to the hyperscaler object store over the Internet.

Context

You edit a secret when its credentials (user name and access token) change.

> ⓘ Note
>
> You can't change the name or the resource group for a secret. If the name or resource group is no longer valid or contains errors, you need to remove the secret and re-create it with the correct details. See Remove an Object Store Secret.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Object Store Secrets*.

   The *Object Store Secrets* screen appears with a tile for each existing secret.
3. Find the tile for the secret and choose *Edit*.

   The *Edit Object Store Secret* dialog box appears.
4. Change the secret details.
   a. Edit the path prefix, if necessary. The path prefix is used to differentiate between different projects which are stored in the same location.
   b. Complete the information requested in the dialog box.
      For S3:
      - Choose `Verify SSL` to apply the SSL security protocol to data transferred from the object store.
      - Choose `Use HTTPS` to apply the HTTPS communication protocol to data transferred from the object store.
   c. Enter the secret in `JSON` format. For more information, see Register Your Object Store Secret
5. Choose *Edit* to save the changes to the secret.

Results

The updated secret appears on the *Object Store Secrets* screen.

6.2.3.2.3   Remove an Object Store Secret

As an administrator, you remove an object store secret when the name is no longer valid or contains errors, or if the secret is no longer required.

Prerequisites

You have the role `aicore_admin_objectstoresecret_editor` or a role collection that contains it. For more information, see Roles and Authorizations.

You have access to the hyperscaler object store over the Internet.

Context

You remove a secret if the name is no longer valid or contains errors, or if the secret is no longer required.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Object Store Secrets*.

    The *Object Store Secrets* screen appears with a tile for each existing secret.
3. Find the tile for the secret and choose *Remove*.

    The *Remove Object Store Secret* dialog box appears.
4. Choose *Remove* to confirm the removal, and remove the secret from your SAP AI Core instance.

Results

The secret no longer appears on the *Object Store Secrets* screen. Any artifacts, such as models or datasets, which are stored in the hyperscaler object store are no longer available for use in your launchpad processes.

6.2.3.3 Manage Applications

You can maintain a list of applications used within your AI processes.

Context

After you have added a Git repository, you'll need to create an application to synchronize your templates in the repository. You use the *SAP AI Core Administration* app to create and manage applications used by your instance of SAP AI Core (your AI runtime).
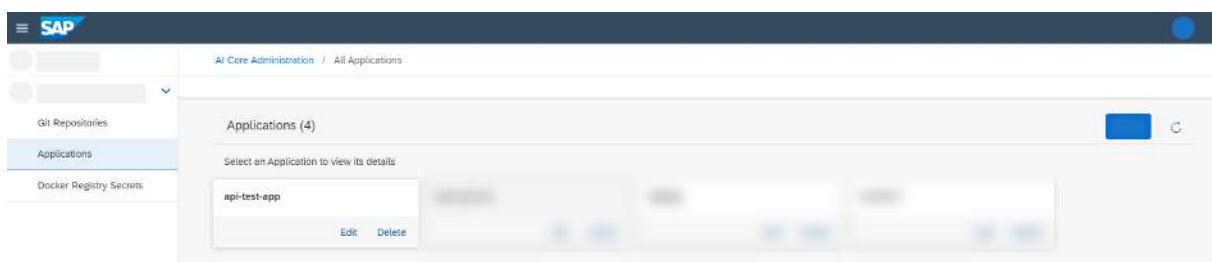
> ⓘ Note
>
> The timing for the initial synchronization is determined by the sync policy for the application. Syncing is not initiated when the application is created in SAP AI Launchpad, as it is managed externally. To check if your application has synced with your Git repository, or to check sync health, you can check the sync data in the application details. See View an Application [page 179].

> ⓘ Note
>
> Users with the `aicore_admin_applications_viewer` role (or a role collection that contains it), can view an application's details but not make changes.

Related Information

Create an Application to Sync your Folders

6.2.3.3.1 Create an Application

As a system administrator, you create applications which sync with Git repositories used in your AI processes.

Prerequisites

- You have the `aicore_admin_applications_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.
- The source Git repository required for the application has been added. See Add a Git Repository.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Applications*.

   The *Applications* screen appears with a tile for each existing application.
3. Choose *Create* to create reference details for a new application.
4. Complete the fields in the *Create a new Application* dialog box as follows:
   a. Enter an application name.
   b. Select a source Git repository.
   c. Enter a path in the repository. The path is used to differentiate between folders within the repository.
   d. Enter a revision value to indicate the specific branch or tag in the repository. For example, **HEAD**.
5. Choose *Create* to create the application for the selected repository.

   The *Applications* screen appears with the new application.

Results

The new application is created.

> ⓘ Note
>
> The timing for the initial synchronization is determined by the sync policy for the application. For SAP AI Core, syncs take place at ~3-minute intervals

Related Information

Create an Application to Sync your Folders

Sync an Application Manually

Prerequisites

You have successfully created an application.

Context

For the SAP AI Core runtime only, automatic syncs occur approximately every three minutes. You can also start the sync manually.

Procedure

1. Navigate to the application's details. See View an Application.
2. Choose *Sync* in the header.

   A dialogue box appears, confirming that your application is being synced.

   > → Tip
   >
   > Choose *refresh* to update the application details.

6.2.3.3.2   View an Application

You can investigate details for an application used within your instance of SAP AI Core.

Prerequisites

You have the `aicore_admin_applications_editor` role or a role collection that contains it. For more information, see Roles and Authorizations [page 321].

Context

Application details include sync status and health information, source repository, and system messages for any synced resources.

Procedure

1. In the *Workspaces* app, choose an AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Applications*.

   The *Applications* screen appears with a tile for each existing application.
3. Find the tile for the application you want to display, and click the tile.

   The *Application Details* screen appears with summary, source, and sync resource details.

   - Summary details include sync status and health (returned by the application), and sync and reconciliation timestamp details

     > ⓘ Note
     >
     > To show extra details for the sync or health status choose 🛈 *(Info)*.

   - Source repository details include the name, URL, repository path, and revision
   - Sync details include resource ID, kind, sync status, and messages for any synced resources

     > ⓘ Note
     >
     > Each synced resource is uniquely identified by an ID. This ID is also visible in the *ML Operations* app for the corresponding executable.

   > → Tip
   >
   > If you have the `aicore_admin_applications_editor` role, you can also *Edit* or *Delete* the application from the *Application Details* screen.

6.2.3.3.3   Edit an Application

As a system administrator, you can edit applications which sync with Git repositories used in your AI processes.

Prerequisites

You have the `aicore_admin_applications_editor` role or a role collection that contains it. For more information, see Roles and Authorizations [page 321].

Context

You edit an application if its source repository, path, or revision details change.

> ⓘ Note
>
> You can't change the name for an application. If the name is no longer valid or contains errors, you'll need to delete the application and re-create it with the correct details. See Delete an Application [page 181].

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Applications*.

   The *Applications* screen appears with a tile for each existing application.
3. Find the tile for the application and choose *Edit*.
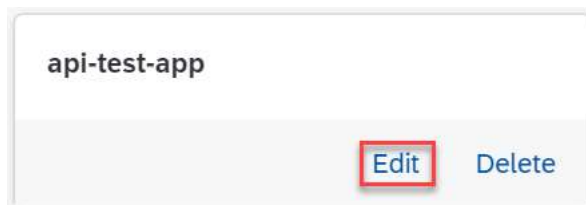
   The *Edit Application* dialog box appears.
4. Change the reference details in the *Edit Application* dialog box as required:
   a. Select an alternative source repository, if necessary. If you choose an alternative repository, you'll need to enter a path and version.
   b. Change the path in the repository.
   c. Change the revision value.
5. Choose *Edit* to save the changes for the application.

   The *Applications* screen appears with the updated application.

6.2.3.3.4 Delete an Application

As a system administrator, you can delete applications which sync with Git repositories used in your AI processes.

Prerequisites

You have the `aicore_admin_applications_editor` role or a role collection that contains it. For more information, see Roles and Authorizations [page 321].

Context

You delete an application if it's invalid or contains errors, or if it's no longer required.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Applications*.

   The *Applications* screen appears with a tile for each application.
3. Find the tile for the application and choose *Delete*.
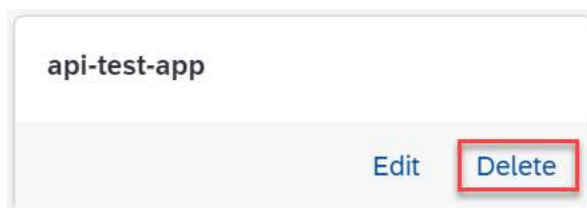
   The *Delete Application* dialog box appears.
4. Choose *Delete* to confirm the deletion, and remove the application from use.

Results

The application no longer appears on the *Applications* screen.

6.2.3.4   Manage Docker Registry Secrets

You can maintain a list of Docker registry secrets for use within your AI processes.

Context

A Docker registry secret holds Docker credentials which enable you to pull Docker images from a Docker registry.

Within SAP AI Core, Docker registry secrets and images are referenced in serving and workflow templates. When an execution or deployment is created using these templates, the credentials are required to read from the Docker registry and pull the associated Docker image.

The example below shows how the Docker secret is used when an ML Operations user creates a deployment.
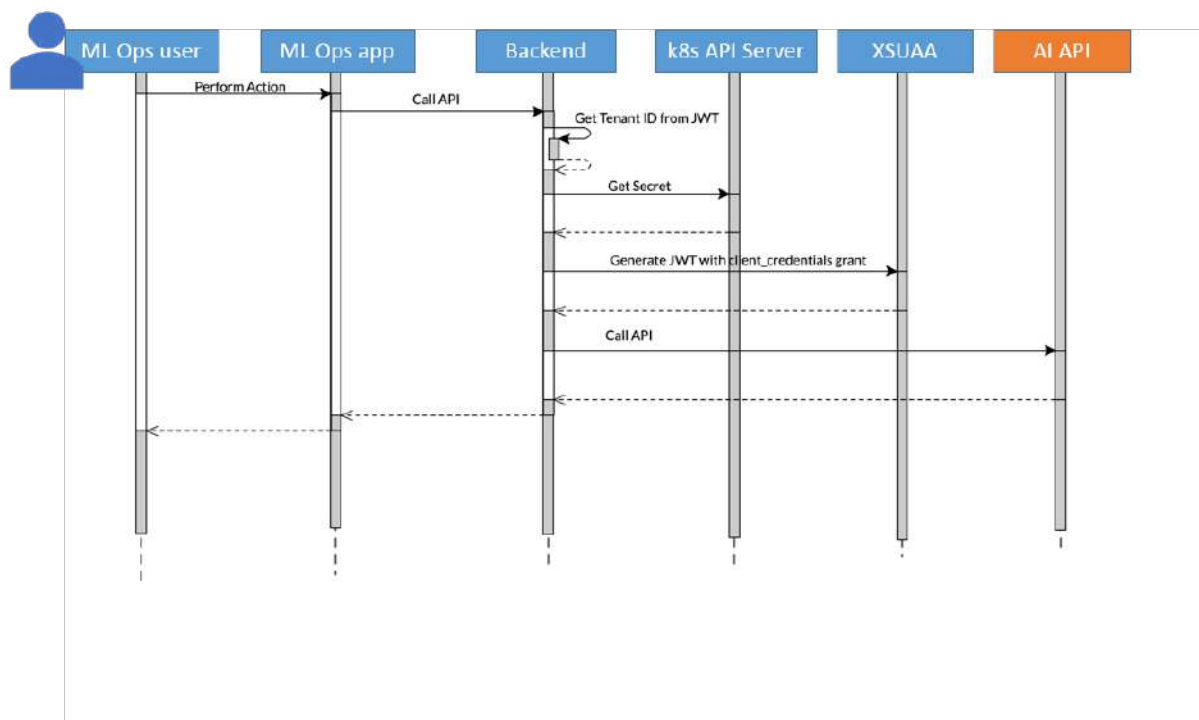
You use the *SAP AI Core Administration* app to add and manage Docker registry secrets. The app supports Docker registry secrets with `Base64` encoding.

Docker registry secrets contain sensitive information. Users with the `aicore_admin_dockerregistrysecret_viewer` role (or a role collection that contains it), can view Docker registry secret names but not credentials.

Related Information

Create Your Docker Registry Secret

6.2.3.4.1 Add a Secret

As a system administrator, you can add Docker registry secrets for use within your AI processes.

Prerequisites

You have the `aicore_admin_dockerregistrysecret_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

You have access to a public-facing Docker registry over the internet. It isn't possible to use a Docker registry behind a VPN or corporate network.

Context

You can use the *SAP AI Core Administration* app to add multiple secrets. The Docker registry must already exist with valid credentials.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Docker Registry Secrets*.

   The *Docker Registry Secrets* screen appears with a tile for each existing secret.
3. Choose *Add* to enter reference details for a new secret.
4. Complete the fields in the *Add a Docker Registry Secret* dialog box as follows:
   a. Enter a name for the secret. This entry is your choice of identifier for your secret.

      Secret names must comply with the following criteria:

      - Contain only lowercase alphanumeric characters, hyphens (-), or periods (.)
      - Start with an alphanumeric character
      - End with an alphanumeric character
   b. Enter the Docker registry secret in `JSON` format.

      Use the secret that has been provided to you by your registry provider. You can copy and paste the secret data, or upload the secret from a file.
      - To copy the secret, copy the contents of the `data` parameter, as provided to you. The secret is between the parentheses. Paste the data into the *Secret* field.
      - To upload the secret as a `.TXT` or `JSON` file, choose ⬆ *(Upload)* . Search for and choose the secret from a local file. The file data then defaults to the *Secret* field.

        > ⓘ Note
        >
        > Files larger than 10 KB can't be uploaded.
5. Choose *Add* to add the secret to your SAP AI Core instance.

Results

The new secret appears on the *Docker Registry Secrets* screen.

The saved credentials enable read access to the Docker registry, and so enable the Docker image to be pulled when an execution or deployment is created.

Related Information

6.2.3.4.2 Edit a Secret

As a system administrator, you can edit Docker registry secrets used within your AI processes.

Prerequisites

You have the `aicore_admin_dockerregistrysecret_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

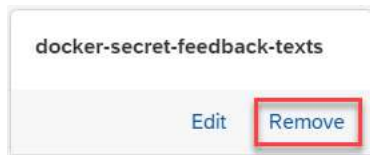You have access to the Docker registry over the Internet.

Context

You edit a secret when its credentials (user name and access token) change.

> ⓘ Note
>
> You can't change the name for a secret. If the name is no longer valid or contains errors, you need to remove the secret and re-create it with the correct details. See Remove a Secret.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Docker Registry Secrets*.

   The *Docker Registry Secrets* screen appears with a tile for each existing secret.
3. Find the tile for the secret and choose *Edit*.
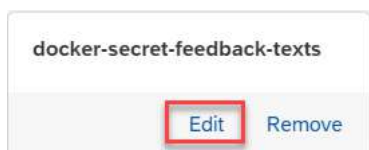
   The *Edit Docker Registry Secret* window appears. The name can't be changed, and the *Secret* field is empty.
4. Enter the secret.

a. To copy the secret, copy the contents of the `data` parameter, as provided to you. The secret is between the parentheses. Paste the data into the *Secret* field.

b. To upload the secret as a `.TXT` or `JSON` file, choose ⬆ *(Upload)* . Search for and choose the secret from a local file. The file data then defaults to the *Secret* field.

> ⓘ Note
>
> Files larger than 10 KB can't be uploaded.

5. Choose *Edit* to save the changes for the secret.

Results

The updated secret appears on the *Docker Registry Secrets* screen.

6.2.3.4.3 Remove a Secret

As a system administrator, you can remove Docker registry secrets used within your AI processes.

Prerequisites

You have the `aicore_admin_dockerregistrysecret_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

You have access to the Docker registry over the Internet.

Context

You remove a secret if the name is no longer valid or contains errors, or if the secret is no longer required.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Docker Registry Secrets*.

   The *Docker Registry Secrets* screen appears with a tile for each existing secret.
3. Find the tile for the secret and choose *Remove*.

docker-secret-feedback-texts

Edit   Remove

The *Remove Docker Registry Secret* dialog box appears.

4. Choose *Remove* to confirm the removal, and remove the secret from your SAP AI Core instance.

Results

The secret no longer appears on the *Docker Registry Secret* screen. Any workflow or serving templates that use the removed secret to create executions or deployments no longer work.

6.2.3.5   Manage Resource Groups

Resource groups are used to physically isolate machine learning workloads, and to logically isolate related resources for a usage scenario.

When your tenant is onboarded, a default resource group is automatically created. Default resource groups can't be deleted.

As an administrator, you create, edit, or delete resource groups, based on your service consumers and usage scenarios.

Runtime entities such as executions, deployments, configurations, and artifacts belong to a specific resource group and aren't shared across resource groups. Scenarios, executables, and Docker registry secrets are shared by all resource groups within a tenant.

A resource group is also referred to as an instance.

> → Remember
>
> Your SAP global account can consist of several accounts. Each account can be associated with a tenant. A tenant can contain multiple resource groups. A tenant always contains a default resource group, as well as the resource groups defined for your usage scenarios.

> ⓘ Note
>
> The maximum number of resource groups is limited at tenant level to 50. If you reach this limit, you receive an error message. To free up space, delete some resource groups. Alternatively, raise a ticket to increase your quota.

Related Information

6.2.3.5.1 Create a Resource Group

As an administrator, you create resource groups to isolate your ML workloads and processes.

Prerequisites

You have the `aicore_admin_resourcegroup_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

Your tenant has been onboarded and a default resource group automatically created.

Context

When creating a resource group, subaccount ID, instance ID, and zone ID are standard labels available for use.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Resource Groups*.

   The *Resource Groups* screen appears with a tile for each existing resource group.
3. Choose *Create* to create reference details for a new resource group.
4. Complete the fields in the *Create Resource Group* dialog box.

Create Resource Group

Resource Group ID*

staging

Labels (1)   +

Key		Value	
ext.ai.sap.com/	Enter label key	Enter label value	✕

subaccount_id

zone_id

instance_id

Create   Cancel

   a. Enter a resource group ID.

> ⓘ **Note**
>
> Ensure that the resource group ID is unique. If the ID isn't unique and is currently in use, the new resource group and its details overwrite the existing resource group.

   b. Choose the `subaccount_id` label key and enter a value.

   c. Choose the `zone_id` label key and enter a value.

   d. Choose the `instance_id` label key and enter a value.

   e. If additional labels are required, enter their keys and corresponding values.

5. Choose *Create* to create the resource group.

   The *All Resource Groups* screen appears and shows the new resource group.

6. **Optional:** Now that the resource group is created, you can link it to an object store secret. To do so, find the resource group and display its details. Choose *Add* to add an object store secret to the resource group. See Add an Object Store Secret.

Results

The new resource group is created. Executions, deployments, configurations, and artifacts that are created using the resource group are uniquely associated with the resource group.

6.2.3.5.2 Edit a Resource Group

As an administrator, you can edit resource groups used within your AI processes. You can change labels and values, or add, edit, or remove object store secrets associated with the resource group.

Prerequisites

You have the `aicore_admin_resourcegroup_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

Context

You edit a resource group when its labels change, or if you need to add or remove related object store secrets.

> ⓘ Note
>
> You can't change the name for a resource group. If the name is no longer valid or contains errors, you must remove the resource group and recreate it with the correct details. See Delete a Resource Group.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Resource Groups*.

   The *Resource Groups* screen appears with a tile for each existing resource group.
3. Choose *Edit* to change reference details for a resource group.

subaccount_id:

zone_id:

instance_id:

Created On:
Oct 11, 2022

4 🏷

Edit   Delete

4. Complete the fields in the *Edit Resource Group* dialog box.
   a. Change the `subaccount_id` value, if necessary.
   b. Change the `zone_id` value, if necessary.
   c. Change the `instance_id` value, if necessary.
   d. If additional label changes are required, then change the label key or corresponding value.
5. Choose *Edit* to save the changes to the resource group.
6. **Optional:** You can also change associated object store secrets. To do so, view the resource group details. Choose *Add*, *Edit*, or *Remove* to change an object store secret for the resource group. See Edit an Object Store Secret and Remove an Object Store Secret.

Results

The resource group is changed. The updated resource group appears on the *All Resource Groups* screen.

6.2.3.5.3 Delete a Resource Group

As an administrator, you delete resource groups which contain errors or which are no longer required in your AI processes.

Prerequisites

You have the `aicore_admin_resourcegroup_editor` role or a role collection that contains it. For more information, see Roles and Authorizations.

Context

You delete a resource group if it's invalid or contains errors, or if it's no longer required.

> ⚠ Caution
>
> You can't delete the default resource group. The default resource group is automatically created when your tenant is onboarded, and it can't be deleted using the *SAP AI Core Administration* app.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. Open the *SAP AI Core Administration* app and choose *Resource Groups*.

   The *Resource Groups* screen appears with a tile for each resource group.
3. Find the tile for the resource group and choose *Delete*.
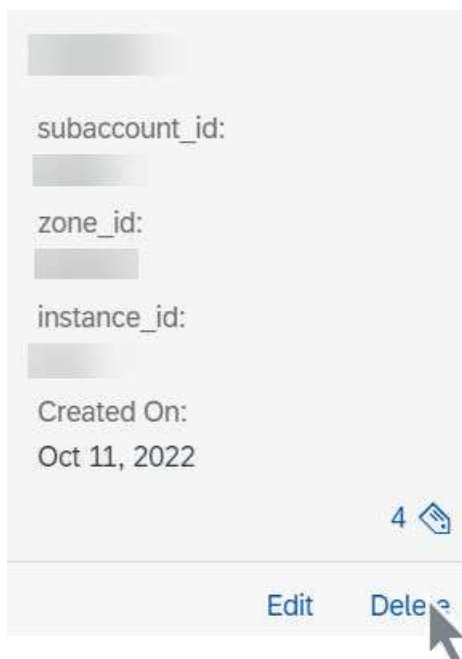
The *Delete Resource Group* dialog box appears.

4. Choose *Delete* to confirm the deletion, and remove the resource group from use.

Results

The resource group is deleted, and any object store secrets added to the resource group are also deleted.

> ⓘ Note
>
> Runtime entities created using the deleted resource group, such as executions and deployments, remain in the launchpad for reference purposes. However, when the associated resource group is deleted, the entities no longer contain a reference to the deleted resource group.

6.2.3.6 Manage Generic Secrets in SAP AI Core

As the main tenant user of the SAP AI Core runtime, you can create, edit, and delete generic secrets at the level of both the main tenant and the resource group. To do so, you use *SAP AI Core Administration* app.

Context

Generic secrets differ from system secrets and can be used to store sensitive information without exposing your credentials.

They are used to store and retrieve secrets, where a system secret is not appropriate.

They are used during AI development both at the tenant level and resource group level.

For more information, see Manage Generic Secrets in SAP AI Core.

6.2.3.6.1 Add a Secret

Prerequisites

- You have the role `aicore_admin_genericsecret_editor` or a role collection that contains it. For more information, see Roles and Authorizations.
- You're using the SAP AI Core runtime.

Context

You use the *SAP AI Core Administration* app to add generic secrets at the level of the main tenant or resource group. For tenant-wide generic secrets, use the SAP AI Core API.

To allow the rotation of tenant-wide secrets for long-running deployments without requiring a restart, the deployment must mount the tenant-wide secret. It must also monitor the mounted secret for changes instead of relying on an in-memory copy. When a tenant-wide secret is updated, the tenant must observe the `resourceGroupSecretReplicationStatus` field in the `Get Secret` endpoint to confirm that the secret has been successfully replicated across the required resource groups. For more information, see Consume Generic Secrets in Executions or Deployments.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. If you want to add your secret at the resource group level, choose the resource group. Alternatively, use the toggles in the header or dialog box, where the system prompts you to specify a resource group.
3. Open the *SAP AI Core Administration* app and choose *Generic Secrets*.

   The *Generic Secrets* screen appears with a tile for each existing secret.
4. Choose *Add* to enter reference details for a new secret.
5. Complete the fields in the *Add Generic Secret* dialog box as follows:
   a. Switch between tenant-level secrets and resource-group-level secrets.
   b. If your secret is at the level of the resource group, confirm the resource group or select a different one by choosing ⊡ *(Change Value)*.
   c. Enter a name for your secret.
   d. Enter the key:value pairs for your secret in one of the following ways:
      - Deselect the *Document Grounding* switch and enter your secret in JSON format.
      - Leave the *Document Grounding* switch selected and choose the document repository type from the dropdown list. The dialog adjusts dynamically for you to fill the remaining information.
      - Leave the *Document Grounding* switch selected and switch to code view (〈·〉), where you can enter your secret in JSON format.

      > ⓘ Note
      >
      > The API expects sensitive data to be Base64-encoded. You can easily encode your data in Base64 format using the following command on Linux or MacOS: `echo -n 'my-sensitive-data' | base64`

   e. **Optional:** Add labels to your secrets. Labels are only required for specific services. They're predefined and are outlined in the related documentation.
6. Choose *Add* to save the secret details.

6.2.3.6.2 Edit a Secret

Prerequisites

- You have the role `aicore_admin_genericsecret_editor` or a role collection that contains it. For more information, see Roles and Authorizations. You can change the secret credentials, but not the secret name or resource group. To change the secret name, delete the existing secret and create a new one.
- You're using the SAP AI Core runtime.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. For secrets at the resource group level, choose your resource group. Alternatively, use the toggles in the header or dialog box, where the system prompts you to specify a resource group.
3. Open the *SAP AI Core Administration* app and choose *Generic Secrets*.
4. Find the tile for the secret and choose *Edit*.

   The *Edit Generic Secret* dialog box appears.
5. Change the secret details. Enter the secret in JSON format. For example:

   ```
   {
           "some-credential": "bXktc2VjcmV0LWNyZWRlbnRpYWw=",
           "other-credentials": "bXktc2VjcmV0LW90aGVyLWNyZWRlbnRpYWw="}
   ```

   > ⓘ Note
   >
   > The API expects sensitive data to be Base64-encoded. You can easily encode your data in Base64 format using the following command on Linux or MacOS: `echo -n 'my-sensitive-data' | base64`

   Add or update your secret labels. Labels are only required for specific services. They're predefined and are outlined in the related documentation.

Results

The updated secret appears on the *Generic Secrets* screen.

6.2.3.6.3   Remove a Secret

Prerequisites

- You have the role `aicore_admin_genericsecret_editor` or a role collection that contains it. For more information, see Roles and Authorizations.
- You're using the SAP AI Core runtime.

Context

You remove a secret if the name is no longer valid or contains errors, or if the secret is no longer required.

Procedure

1. In the *Workspaces* app, choose the AI API connection.
2. For secrets at the resource group level, choose your resource group. Alternatively, use the toggles in the header or dialog box, where the system prompts you to specify a resource group.
3. Open the *SAP AI Core Administration* app and choose *Generic Secrets*.

   The *Generic Secrets* screen appears with a tile for each existing secret.
4. Find the tile for the secret and choose *Remove*.

   The *Remove Generic Secret* dialog box appears.
5. Choose *Remove* and confirm the removal.

Results

The secret no longer appears on the *Generic Secrets* screen.

6.2.4 ML Operations

This section outlines the different components of the *ML Operations* app and how to use them for lifecycle management of a scenario.

The *ML Operations* app helps you manage the lifecycle tasks for an AI use case (business project), created for a resource group that exists on an AI runtime platform such as SAP AI Core.
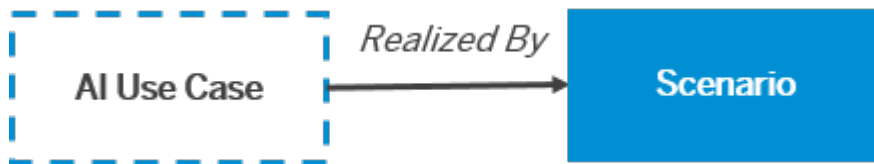
A use case is a scoped project to realize business value through AI technology.

> **✤ Example**
>
> In the Example End-to-End Use Case [page 318], a hypothetical airline company has an AI use case relating to **feedback** classification (complaints versus compliments). The business use case involves classifying feedback (texts) as compliments or complaints.

A scenario is a technical realization of (part of) a use case benefitting from using AI technology.

> **✤ Example**
>
> In the Example End-to-End Use Case [page 318] to classify feedback, the `text-clf-tutorial-scenario` scenario is used.
>
> #
>
>

The components within the *ML Operations* app are used to complete lifecycle tasks for AI scenarios.
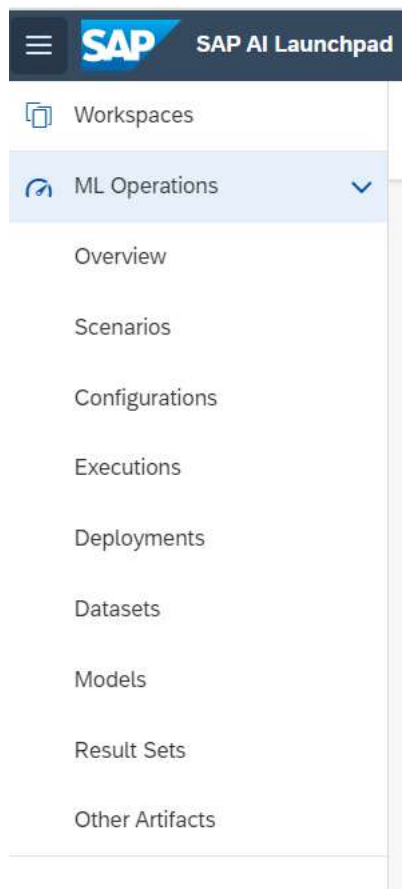
Use the *ML Operations* app components to complete the following:

Components Used in AI Lifecycle

Lifecycle Tasks for a Scenario (AI use case)	Component in *ML Operations* app
Set and track values (dataset and parameters) for training a model	Configurations
Train and generate an AI models using configuration settings	Executions
Select the most recently trained model for a deployment pipeline	Configurations
Deploy the AI model and generate endpoints for consumption using configuration	Deployments

The *ML Operations* app is used to complete lifecycle tasks for the scenarios available in your selected AI API connection (and the selected resource group). Your actions do not change other resource groups (present within same AI API connection), even if the same scenario is shared by different resource groups. For more information, see Set Resource Group [page 164].

Prerequisites

A valid AI API connection is required before you can access the *ML Operations* app. To create a connection to the underlying AI runtime, see Add Connection to SAP AI Core [page 160].

Accessing the App

You access the *ML Operations* app within SAP AI Launchpad via the side navigation panel.

To access the app:

1. Select the connection to your runtime in the *Workspaces* app.
   The *ML Operations* app is now visible in your side navigation panel, and resource groups are listed.
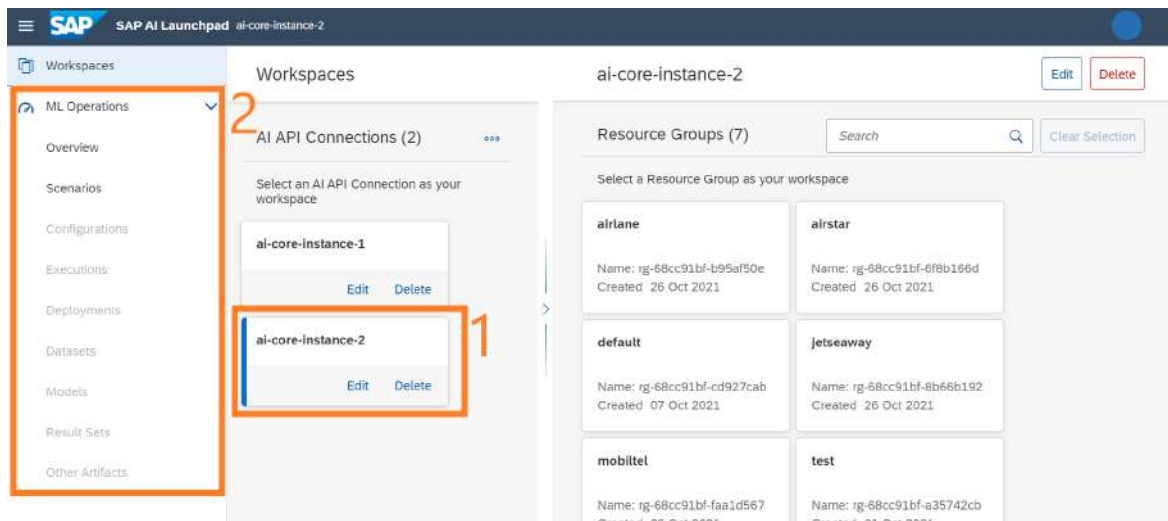
2. Select the resource group from the *Resource Group* pane in the *Workspaces* app. This enables access to the components within the *ML Operations*.
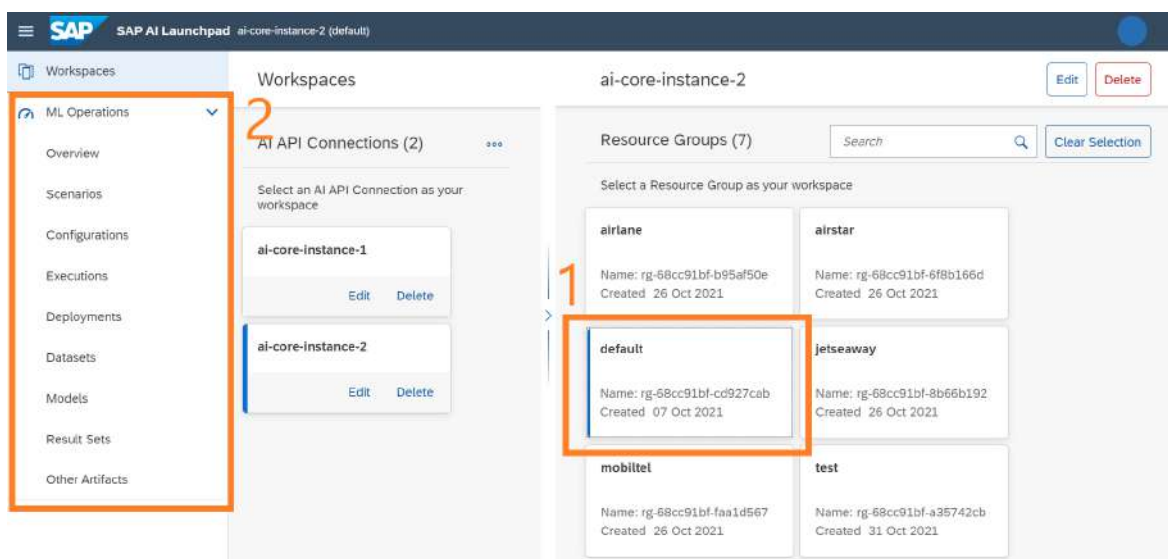
> ⓘ Note
>
> When you select an AI API connection (to your AI runtime) as your workspace, you'll see that the *ML Operations* app becomes accessible and the components for *Configurations*, *Executions*, *Deployments*, *Datasets*, *Models*, *Result Sets* and *Other Artifacts* are disabled in the side navigation panel. This is because access to these components is tied to a resource group. You'll need to select a resource group within your workspace to access these components (demonstrated in images above). For more information, see Set Resource Group [page 164].

> ⓘ Note
>
> If some components are still inaccessible after selecting a resource group as your workspace, then you do not have the access rights required to view these components. Contact your administrator to grant you the required roles.

Role Requirements

ML Operations app users should be assigned the following roles:

- `operations_manager` to access the *ML Operations* app in the side navigation panel
- `mloperations_viewer` to view app content, or
- `mloperations_editor` to view and change app content

Related Information

Set Resource Group [page 164]
Custom Access for Resource Groups [page 333]

6.2.4.1 Overview

The *ML Operations* app provides an overview of statistics for your scenarios within your SAP AI Core (AI runtime), used by your selected connection.

The *Overview* provides a dashboard for you to visualize the following:

- Main tenant charts
  Visualization of all the scenarios (created for all resource groups) in your SAP AI Core (AI runtime). For more information, see Resource Groups [page 44].
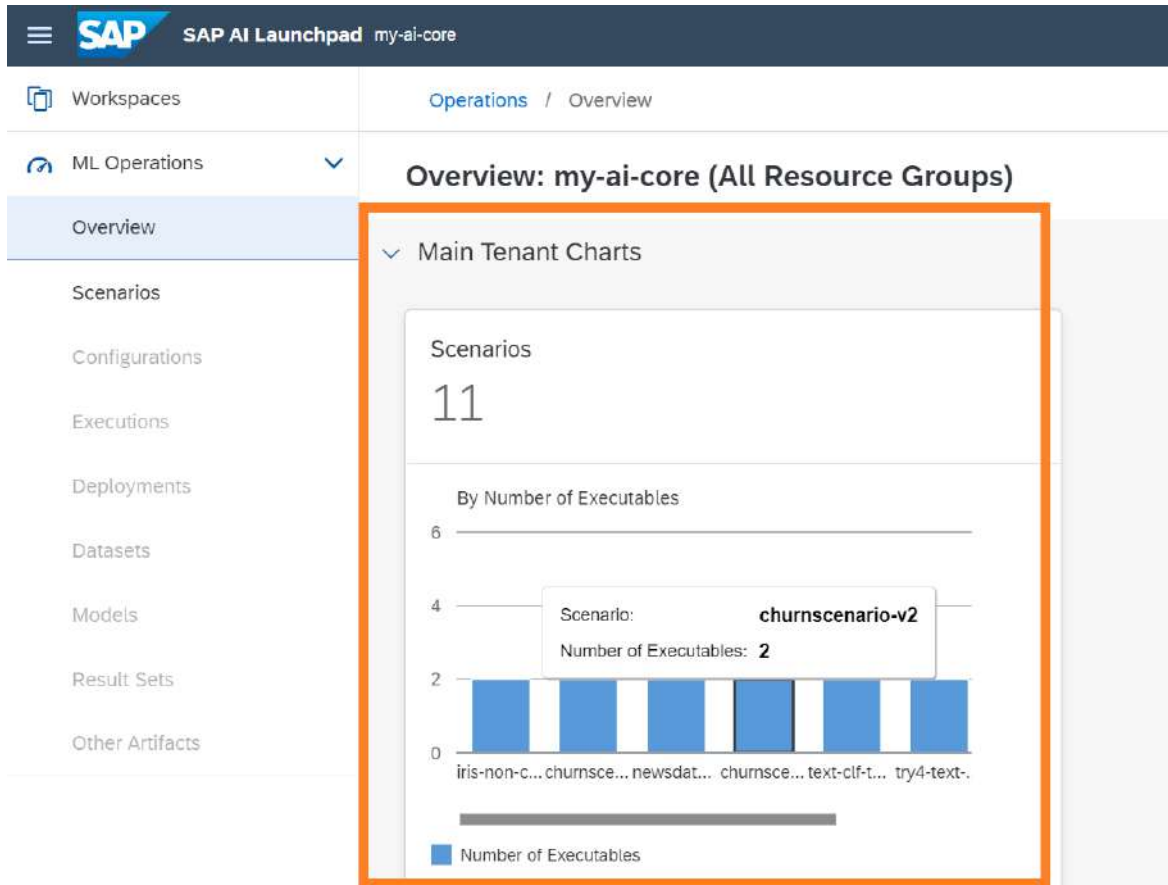
- Resource group charts
  Visualization of the following:

  - Executions: number of trainings (across all resource groups for the selected connection)
  - Deployments: number of active deployments (across all resource groups for the selected connection)
  - Artifacts: number of registered datasets and models (across all resource groups for the selected connection)
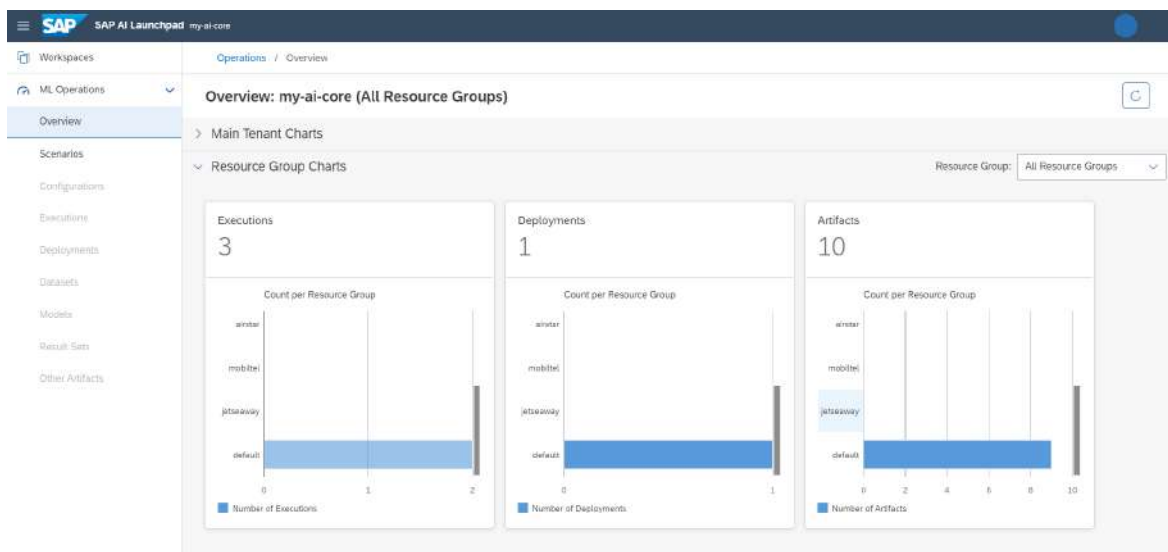
6.2.4.2 Scenarios

A scenario in the *ML Operations* app is a collection of executables.

A scenario is a technical realization of a business AI use case (for example, a recommendation system, a review classifier).

An executable refers to a template for an AI pipeline, which means there are placeholders to set values in AI pipelines. An AI pipeline is a piece of code that trains AI models or deploy those models. A scenario is a group of executables (AI pipelines) to realize an AI use case.

A scenario is a collection of the following kinds of executables:

Workflow executable	An executable that trains an AI model. A file (script) to read datasets and generate a model after training. It can also be used for batch inferencing. For more information, see Workflow Templates.
Serving executable	An executable that serves (deploys) an AI model. A file (script) to load an AI model and serve it for inferencing (making predictions). For more information, see Serving Templates.

Scenario(s) and their contained executables are available across all resource groups. For more information about how to isolate datasets and models for the same scenario, see Resource Groups.

6.2.4.2.1 View a Scenario

You use the *ML Operations* app to list all scenarios for a selected connection.

Prerequisites

You have either the `mloperations_viewer` or `scenario_metadata_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Select an AI API connection (setting resource group isn't required). For more information, see Set Resource Group.

2. Choose the *Scenarios* tile on the *Overview* dashboard, or in the *ML Operations* app, choose *Scenarios*.

   The *Scenarios* screen appears, listing all the scenarios for the selected connection. Scenarios are listed by name, and with additional details such as description, created on timestamp, changed on timestamp, number of contained executables, and number of versions.

3. To view the details for a scenario, select a scenario in the list or choose ⟩ *(More)*. The *Scenario Details* screen appears with the *Overview* tab displayed. The overview shows a graphical summary of the scenario, showing the distribution of workflow executables and serving executables by scenario version.

4. **Optional:** To view your executables. choose the *Executables* tab to list the workflow and serving executables for the scenario across its various versions.

   - Filter the list by choosing a version in the *Show* field, or show all versions.
   - Search the list by entering a name, partial name, or part of the description in the Q *(Search)* field.
   - Display the details for an executable by selecting an executable in the list, or choosing ❯ *(More)*.

5. **Optional:** To view your prompt templates, choose the *Prompt Templates* tab.

   - Filter your prompt template using the filter field.
   - Choose a prompt template to see its detailed view.
     - You can delete a template revision using the 🗑 *(Delete)* icon.
     - You can delete a template by deleting all of its revisions.
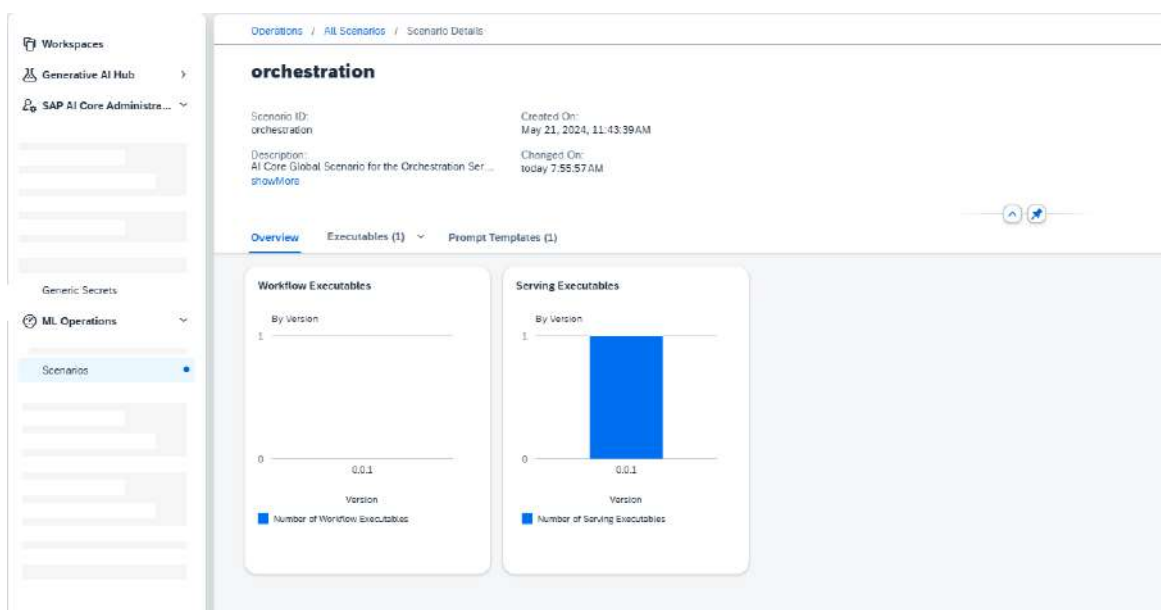
Related Information

6.2.4.2.2 Executables

An executable is used to define training or serving pipelines for an AI use case. Executables for the same AI use case are grouped by scenario.

> ❖ Example
>
> In SAP AI Core, an executable can be a template. For more information, see Templates.

- An executable that is used to train an AI model is called a workflow executable.
- An executable that is used to serve (deploy) an AI model is called a serving executable.

Executables contain any number of placeholders, such as parameters (for hyperparameters of the AI model), input artifacts (for datasets, models), and output artifacts (for models). Placeholders make it easy for resource groups to implement the executable with their own datasets or parameters.

- To display an executable, navigate to the scenario details screen and click on the executable's name. For more information, see View a Scenario [page 202].
- To use an executable, you combine values to input artifacts using a configuration. In a configuration, you set values for these placeholders. You can then start the training or deploying process by creating a instance using that configuration.
  You can also define *Labels* that can be associated with an executable to provide annotations. These labels are listed on the executables details screen along with *Input Artifacts*, *Parameters*, and *Output Artifacts*.
- To use executables, you have either the `mloperations_viewer` or `scenario_executable_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations [page 321].

> ✤ Example
>
> Consider a scenario, such as a Product Review Classification, involving positive or negative reviews.
>
> This scenario has a workflow executable named **`Product_Review_Train`** with the following placeholders:
>
> - Input parameters
>   - `Num_of_epochs`
>   - `Max_Depth_Decision_Tree`
> - Input artifacts
>   - `Stopwords_Dataset`
>   - `Past_Reviews`

Related Information

Executables [page 203]
Serving Executables [page 206]

6.2.4.2.2.1 Workflow Executables

An executable that is used to train an AI model or perform batch inferencing is called a workflow executable.

Like any other executable, a workflow executable contains placeholders for input artifacts (datasets in most cases) and parameters. The values for these placeholders are provided by creating a configuration.

> ✤ Example
>
> In SAP AI Core, a workflow executable can be an Argo workflow template. For more information, see Workflow Template.

The following is an example of a workflow executable.

- The scenario name is included in the menu path.
- The executable name, executable ID, description, and version are displayed in the header.
- The *Deployable* field indicates whether the executable is intended for training or deployment purposes. A `No` value indicates that the executable is intended for training.
- The *Input Artifacts* section contains the dataset.
- The *Output Artifacts section* contains the name of the model that will be generated.

Workflow steps:

1. Create a configuration to combine a dataset with a placeholder (input artifact), and set values (integer, string or float) for the parameter placeholders of the workflow executable.
2. Start the training process by choosing *Create Execution* on the configuration details screen.
3. Upon completion, the execution results in an AI model with the same name as the output artifact mentioned in the workflow executable.

> ⓘ Note
>
> SAP AI Launchpad is an interface to your AI runtime, therefore the workflow executable (scenario inclusive) is present in your selected runtime connection.

Related Information

Executions [page 212]
Configurations [page 207]

6.2.4.2.2 Serving Executables

An executable that is used to deploy (serve) an AI model is called a serving executable.

Like any other executable, a serving executable contains placeholders for input artifacts (usually models), parameters (hyperparameters for the model) and output artifacts (although rarely used with serving executables). The values for these placeholders are provided by creating a configuration. You use the serving executable and configuration to create a deployment for online predictions.

> ⚬ Example
>
> In SAP AI Core, a serving executable can be a template. For more information, see Serving Template.

The following is an example of a serving executable.
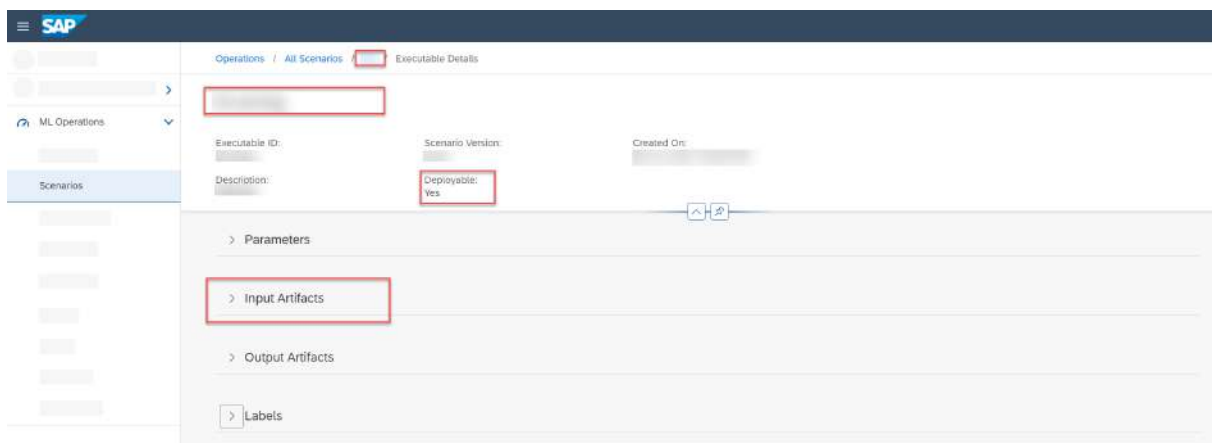
- The scenario name is included in the menu path.
- The executable name, executable ID, description, and version are displayed in the header.
- The *Deployable* field indicates whether the executable is intended for training or serving purposes. A `Yes` value indicates that the executable is intended for deployment (serving).
- The *Input Artifacts* section contains the dataset.

Workflow steps:

1. Create a configuration to combine a model (artifact) with a placeholder (input artifact), and set values (integer, string or float) for the parameter placeholders of the serving executable.
2. Start the deployment (serving) process by choosing *Create Deployment* on the configuration details screen.
3. Upon completion of a deployment, an endpoint is generated. The endpoint can be used in real-time for your AI use case.

> ⓘ Note
>
> Since SAP AI Launchpad is an interface to your AI runtime instance, the serving executable is also present in your AI runtime.

Related Information

6.2.4.3   Configurations

Configurations combine artifacts (such as datasets or models) with executables, so that training or deployment processes can be undertaken.

Configurations combine the following:

- Executables and scenarios
  Executables and scenarios include the specific version of the AI pipeline template that is used to train or deploy an AI use case. They contain placeholders for artifacts and parameters.
- Artifacts
  Artifacts are either datasets (used for training), or models (used for deployment). Artifacts fill the placeholders in an executable (input artifacts and output artifacts).
- Parameters (manually supplied)
  Parameters are alphanumeric values that are used mostly for hyperparameters. Hyperparameters are variables that change a model's decision-making capabilities. They are used to fill the placeholders in an executable.

You can create multiple configurations using the same or different values. After you have defined your configuration and defined the values, you can start the process of training or deployment.

Configurations can be reused in multiple executions, where data `kinds` match.

Related Information

6.2.4.3.1   View a Configuration

A configuration consists of parameters and input artifact references for training or deployment processes.

Prerequisites

You have either the `mloperations_viewer` or `scenario_configuration_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Choose the resource group. For more information, see Set Resource Group. Note that configurations are not shared across resource groups.

2. In the *ML Operations* app, choose *Configurations*.

   The *Configurations* screen appears listing all of the configurations for the selected resource group. The columns include the following information:
   - *Name / ID*: name and ID of the configuration
   - *Scenario*: name of the scenario which uses the configuration
   - *Executable*: name of the executable (template) used in the configuration
   - *Created on*: timestamp indicating when the configuration was created
   - *Parameters* and *Input Artifacts*: number of parameters (alphanumeric values) and input artifacts (dataset or model) used by the configuration

3. **Optional:** Search the list by entering a configuration name or part of the name in the Q *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

4. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.
   a. Enter a configuration ID. A valid configuration ID consists of lowercase characters, hyphens (-), and numbers.
   b. Alternatively, select a scenario and further choose attributes, such as version and executable.
   c. Choose *Apply* to apply the filter and conditions to the list.

5. To view the details for a configuration, select a configuration in the list or choose ❯ *(More)*. The *Configuration Details* screen appears. The overview shows a summary of the configuration.
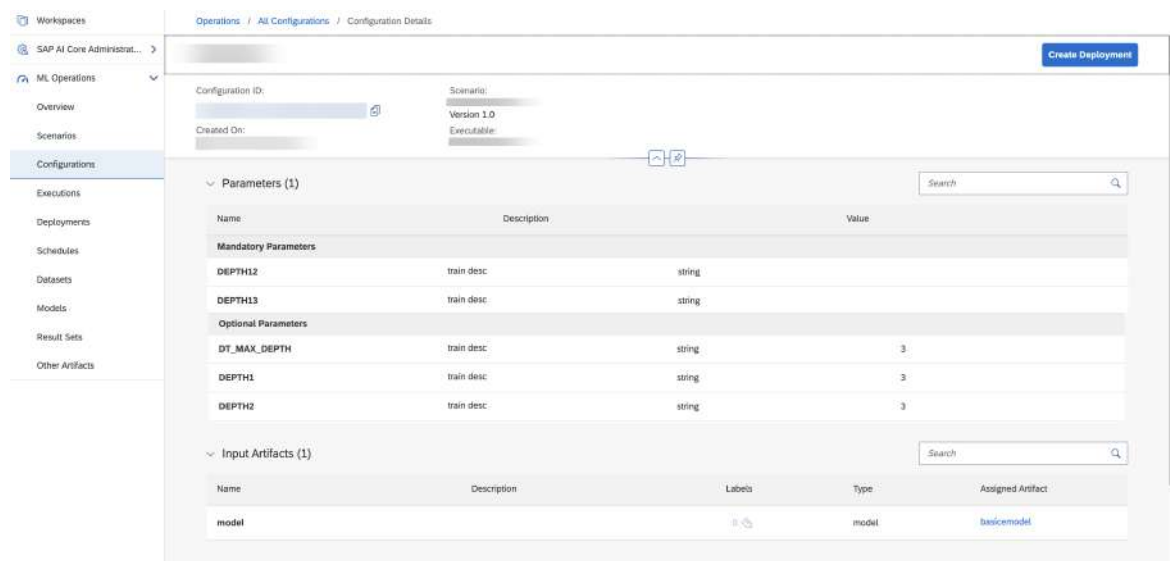
6. **Optional:** Create either an execution or deployment using the configuration.
   a. When a configuration is used for training, choose *Create Execution* to initiate the training process.

b. When a configuration is used for deploying a model, choose *Create Deployment* to deploy the model.

7. **Optional:** Choose an artifact to redirect to a list of artifacts, prefiltered based on attributed of the chosen artifact.

Related Information

6.2.4.3.2 Create a Configuration

A configuration combines parameters, artifacts (for example, a dataset or model), and executables, that are used in training or deploying a model.

Prerequisites

You have either the `mloperations_editor` or `scenario_configuration_editor` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. In the *ML Operations* app, choose *Configurations*.
2. On the *Configurations* screen, choose *Create*.

   The create wizard appears. This wizard has two main areas; a navigation area at the top showing the step sequence, and a content area below it.
3. Enter the configuration details.

   Required details include the following:
   - *Configuration Name*: enter a name for the new configuration
   - *Scenario*: enter a name which represents the AI use case
   - *Version*: choose a scenario version
   - *Executable*: select either a workflow or serving executable from the list displayed. Selection options are subject to availability and the selected version.
     - *Workflow Executable* for a training template
     - *Serving Executable* for a serving (deploying) template

4. Choose *Next* to display the *Input Parameters* step.

5. Enter an alphanumeric value for the *Input Parameter*, as required by placeholder (`<Parameter>`) of the selected executable. Where default values have been set, these will be prefilled, but can be changed manually.

> ⚠ Restriction
>
> A configuration can contain up to 1000 input parameters. An input parameter value can't exceed 5000 characters.

> → Tip
>
> The *Reset* button clears the mandatory fields and enters default values, where they are assigned. The *Enable Description* switch show descriptions and types, where available.

6. Choose *Next* to display the *Input Artifacts* step.

7. Select the required artifacts from the list of available artifacts. Where possible, the artifacts available have been prefiltered by type. The artifact is required as an input artifact for the selected executable.

> → Tip
>
> Use the *Enable Description* switch to view artifact description as popovers.
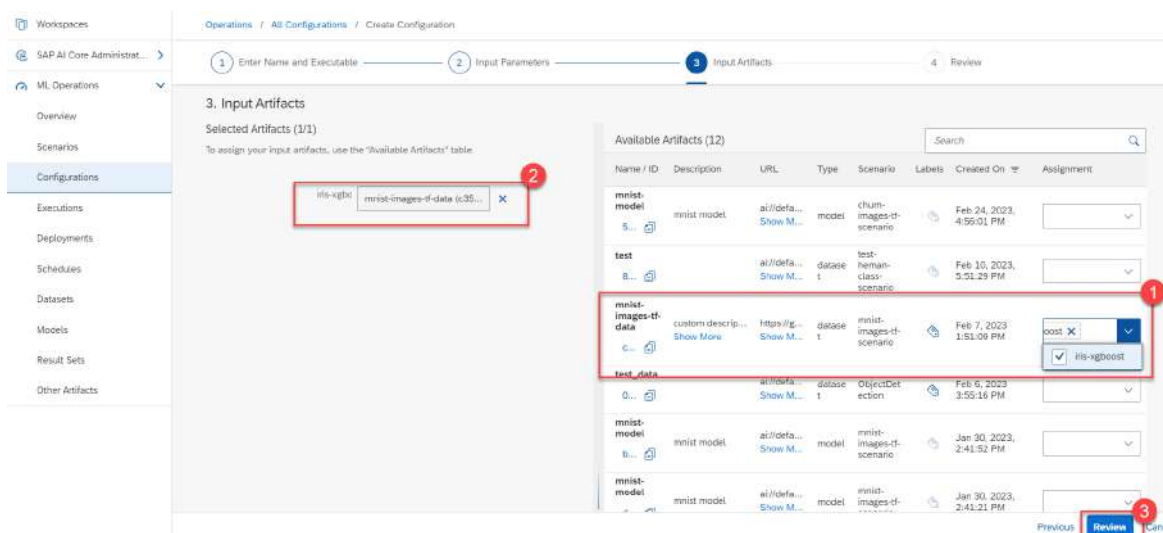
8. Choose *Review* to check the details you've entered for the configuration. Check the details then choose *Create*.

   You'll be redirected to the details screen for the newly created configuration.

9. **Optional:** Choose ▤ *(Copy)* for the configuration ID to copy the unique ID for future use.

Results

The newly created configuration is now available on the *All Configurations* screen.

Related Information

Create a Deployment [page 230]
Create an Execution [page 213]

6.2.4.4 Executions

An execution is a training process or batch inferencing process for an AI scenario. The behavior of an execution is determined by the code pieces in the workflow executable.

A workflow executable is an AI pipeline that trains a model or generates batch inferencing. You use a configuration to set values (parameters and datasets) for a workflow executable. The configuration is then used to create an execution.

Depending on the code in the workflow executable, an execution accomplishes one of the following:

- Trains and generates an AI model. For more information, see Models (artifacts) [page 252].
- Generates batch inference (predictions on subsets of data). For more information, see Result Sets (Artifact). [page 260]

Multiple executions can be created using the same configuration. An execution is an instance of workflow executable (used in the configuration). For more information, see Configurations [page 207].

> ⓘ Note
>
> An execution runs only once and is not reusable. However, you can run another execution using the same configuration, and the same combination of values for workflow executable and dataset.

Custom Features

> → Remember
>
> The features that you see in the user interface depend on the capability settings in your underlying runtime/ service. For more information, see Custom Runtime Capabilities Using the Meta API [page 43].

Execution Statuses

Executions can have any of the following statuses:

Pending
Running
Stopping
Stopped
Completed
Dead
Unknown

The following figure shows how execution states can change following the initial status of *Pending*:
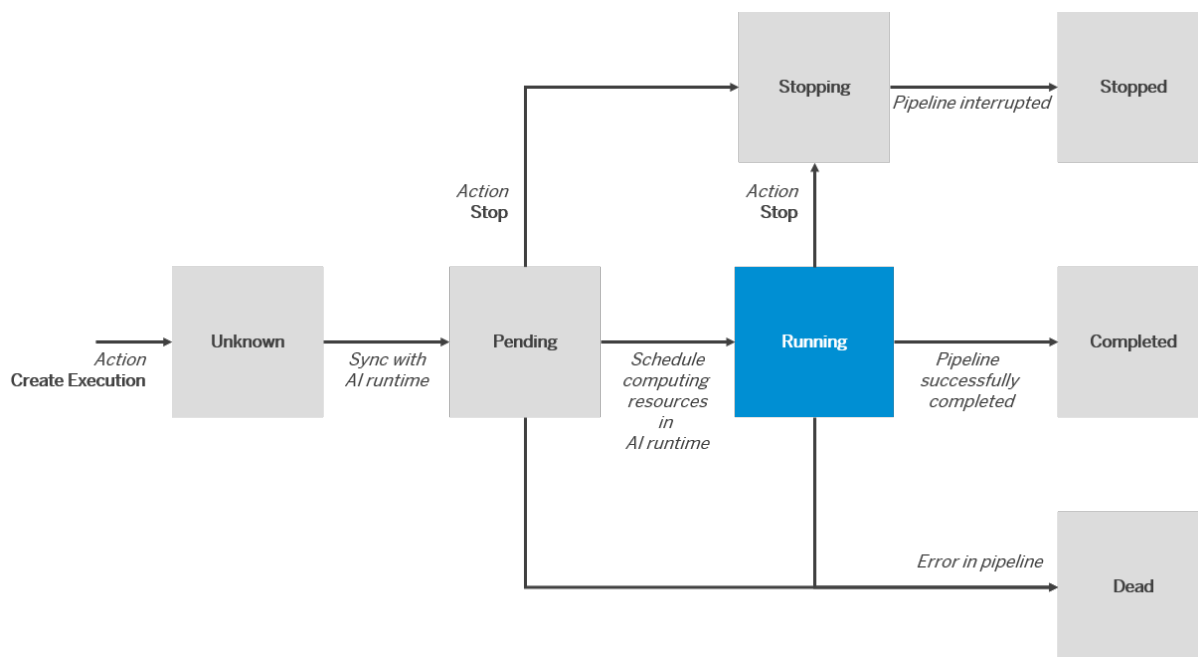
Execution Status Flow

When an execution leaves the *Running* status, the computing resources that were used by the AI runtime are released.

Stop/ Delete Behavior by Status for Executions

Status	Stop Execution	Delete Execution
Unknown	Not enabled	Enabled
Pending	Enabled	Not enabled
Running	Enabled	Not enabled
Completed	Not enabled	Enabled
Stopping	Not enabled	Not enabled
Stopped	Not enabled	Enabled
Dead	Not enabled	Enabled

6.2.4.4.1 Create an Execution

Prerequisites

You have the `mloperations_editor` or `scenario_execution_editor` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Configurations* and create a configuration for a workflow executable.

   For more information, see Create a Configuration.
3. Navigate to the configuration details. See View a Configuration.
4. Choose *Create Execution*. The model training then starts.
5. **Optional:** When the *Create Execution* dialog appears, search the list by entering a configuration name or part of the name in the 🔍 *(Search)* field.

> → Tip
>
> When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.
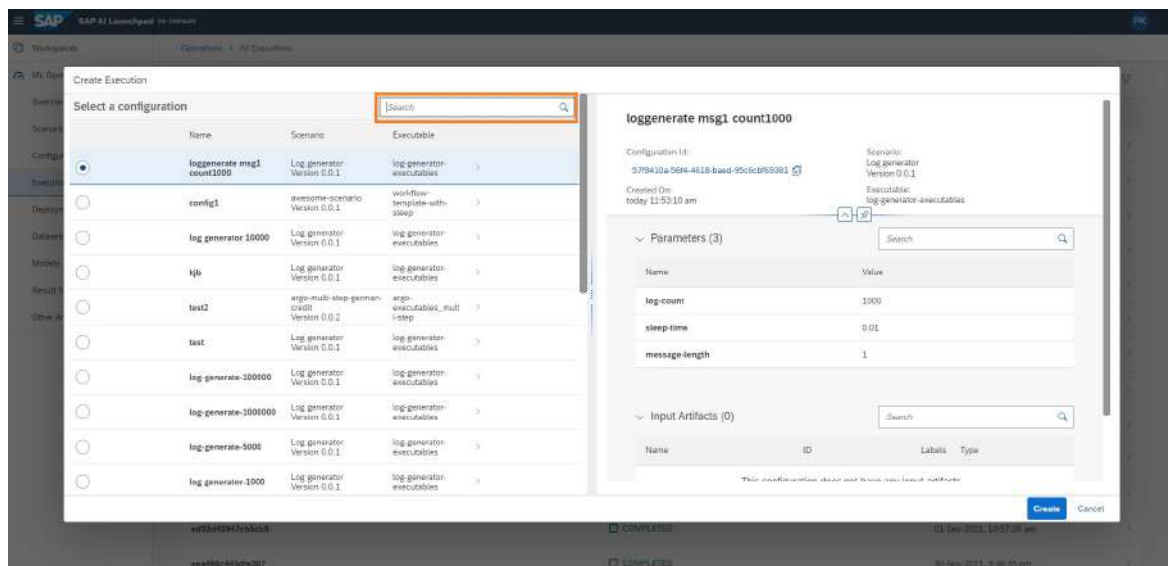
> ⚠ Restriction
>
> The number of pods used by executions is limited at tenant level. A tenant is allowed to have at least 50 pods before the quota is enforced. If your tenant reaches this limit, your execution will be queued. You can raise a ticket to increase your quota.

6.2.4.4.2 View an Execution

You can view the details for an execution, and investigate details for each operation in the lifecycle process. You can use the overview like a dashboard to explore an execution's operations.
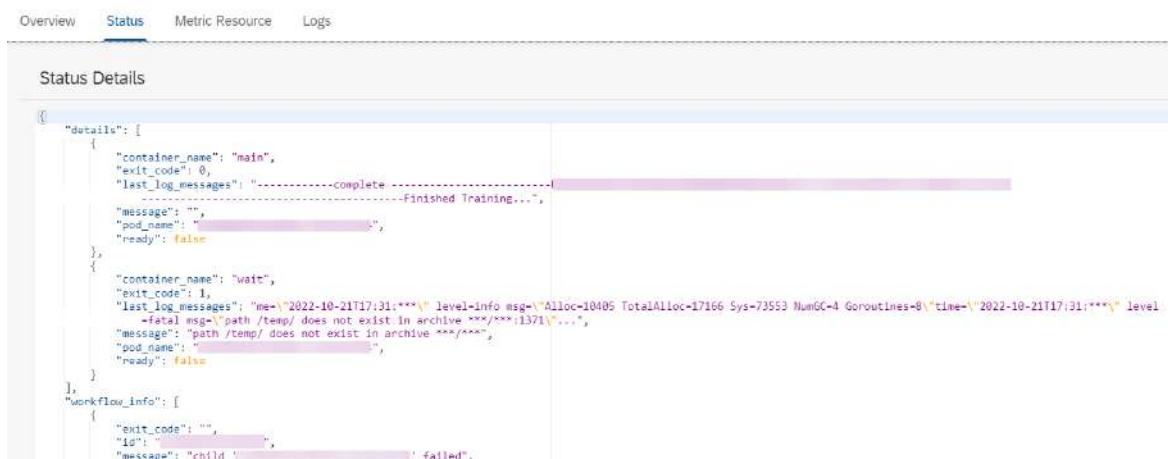
Prerequisites

You have either the `mloperations_viewer` or `scenario_execution_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Executions*.

   The *Executions* screen appears listing all of the executions for the selected resource group. Executions are listed by ID, and with additional details such as configuration name and ID, current and target status, created on timestamp, and changed on timestamp.

3. **Optional:** Filter the list by choosing ∇ *(Filter)*. The *Filter* dialog appears.
   a. Enter an execution ID, if required. A valid execution ID consists of lowercase characters, hyphens (-), and numbers.
   b. Alternatively, enter a configuration ID or status, or choose a scenario and additional attributes, such as version and executable.
   c. Choose *Apply* to apply the filter and conditions to the list.

4. To view the details for an execution, select an execution in the list or choose > *(More)*.

   You'll see an overview of the execution, including the overview steps, and tabs with more information.
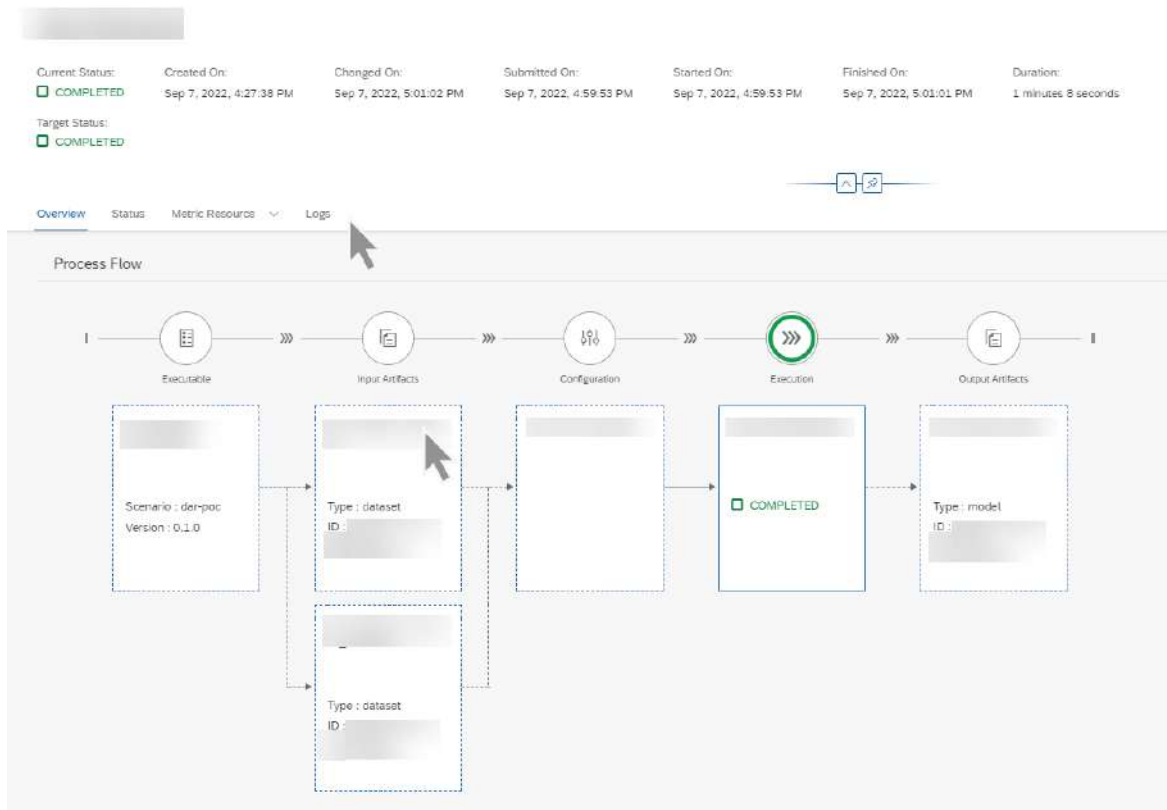
> → **Tip**
>
> Timestamp details for each step in the execution process can be seen in the header. These dates and times show when the execution was created or changed. Timestamps are displayed in the user's local time zone.
>
> The *Overview* tab shows the operations involved in the execution, as well as summary data, such as input dataset and output model (training) or result set (inferencing). You can click each card in the process to display the specific operation in more detail.

5. **Optional:** Choose the *Status* tab to check detailed message, status, and severity information for the execution. See View Status Details.

6. **Optional:** Choose the *Metric Resource* tab to check the metrics and custom information for the execution. See View the Metric Resource for an Execution.

7. **Optional:** Choose the *Logs* tab to check or download the logs for the execution. See View Execution Logs.

Related Information

Compare Executions [page 223]

6.2.4.4.3 View Status Details

You check the status details for an execution to see detailed code logs for a running execution. The status details include detailed status and workflow information.

Prerequisites

You have either the `mloperations_viewer` or `scenario_execution_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

Once an execution is created, the status details provide information about the running execution.

Procedure

1. Navigate to the execution's details. See View an Execution.
2. Choose the *Status* tab to view status details. You may need to scroll to check all status details for the execution.

6.2.4.4.4 Stop an Execution

Stopping an execution releases the computing resources acquired in the AI runtime in which the execution is present (such as SAP AI Core).

Prerequisites

You have one of the following roles:

- `mloperations_editor`
- `scenario_execution_editor`

For more information, see Roles and Authorizations.

Context

You can stop an execution if it has a *Running* or *Pending* status. For executions in other statuses, the *Stop* button is not enabled.

Procedure

1. Navigate to the execution's details. See View an Execution.
2. Choose *Stop* in the header. A *Warning* dialog box appears.
3. Choose *Stop* to stop running the execution.

Results

The execution is stopped, and the computing resources acquired for the execution are released.

6.2.4.4.5 Stop Multiple Executions

Prerequisites

You have the `mloperations_editor` or `scenario_execution_editor` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

- The maximum number of updates per request is 100.
- Only *stopped*, *dead* or *unknown* executions or deployments can be deleted.
- Only *running* or *pending* executions or deployments can be stopped.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Executions*.

   The *Executions* screen appears listing all of the executions for the selected resource group. Executions are listed by ID, and with additional details such as Name, Current and Target Status, Created On timestamp, and Changed On timestamp.
3. Choose the Executions you want to stop using, the checkboxes.
4. Choose *Stop* in the header.

Results

A dialog box appears, confirming how many of your selected executions will be stopped.

> ⓘ Note
>
> Where executions or deployments with a mixture of states that can be stopped **and** deleted are selected, only items eligible to the chosen request will be successfully processed.

6.2.4.4.6 Delete an Execution

You delete an execution to remove it from your instance. Deletion does not impact the associated model or result sets.

Prerequisites

You have the `mloperations_editor` or `scenario_execution_editor` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

You can delete an execution if it has the status *Completed*, *Dead*, *Stopped*, or *Unknown*. For executions in other statuses, the *Delete* button is not enabled.

Procedure

1. Navigate to the execution's details. See View an Execution.
2. Choose *Delete* in the header. A *Warning* dialog box appears.
3. Choose *Delete* to confirm the deletion.

Results

The execution is deleted and is no longer available for selection.

> → Remember
>
> Deleting an execution doesn't delete the associated model or result sets.

6.2.4.4.7 Delete Multiple Executions

Prerequisites

You have the `mloperations_editor` or `scenario_execution_editor` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

- The maximum number of updates per request is 100.
- Only *stopped*, *dead* or *unknown* executions or deployments can be deleted.
- Only *running* or *pending* executions or deployments can be stopped.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Executions*.

   The *Executions* screen appears listing all of the executions for the selected resource group. Executions are listed by ID, and with additional details such as Name, Current and Target Status, Created On timestamp, and Changed On timestamp.
3. Choose the Executions you want to stop, using the checkboxes.
4. Choose *Delete* in the header.

Results

A dialog box appears, confirming how many of your selected executions will be deleted.

> ⓘ Note
>
> Where executions or deployments with a mixture of states that can be stopped **and** deleted are selected, only items eligible to the chosen request will be successfully processed.

> → Remember
>
> Deleting an execution doesn't delete the associated model or result sets.

6.2.4.4.8 View the Metric Resource for an Execution

Use metric data to determine a model's quality (accuracy, precision, or any other custom metric value).

Context

Quality metrics (associated labels, tags and custom info) for a model are generated by the workflow executable used in the execution (training process). Metrics differ between executions, and depend on the input artifacts (dataset) and parameters specified in the configuration.

For more information about how to generate metrics during an execution, see Storing Metric Data.
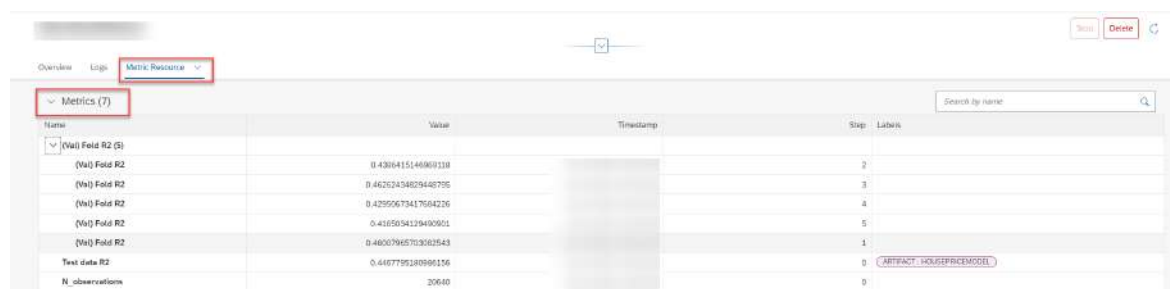
Procedure

1. In the *ML Operations* app, find an execution and display its details. For more information, see View an Execution.
2. To view the metrics resource data, select the *Metric Resource* tab.

> ⓘ Note
>
> The *Metric Resource* tab is only visible if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

The metrics generated for the execution (training process) are displayed.

Metric details are displayed, as follows:

- *Name*: Quality criteria, for example, accuracy or mean absolute error (MSE).
- *Value*: Indicates a level of quality, and is dependent on the *Name* (criteria).
- *Timestamp* and *Step*: Uniquely identify or differentiate the results. For metrics that are logged multiple times, the timestamp and step can be used in conjunction to check how a metric has progressed during the execution (training process). For example, a model trains iteratively on same dataset in a single training process. Also known as an epoch.
- *Labels*: Classifying phrase/ name applied to the metric for that training pipeline template (executable).

> ⓘ Note
>
> The same data can be viewed from the details screen of the generated model, see View Metrics for a Model.

3. To display the names and key/ value pairs associated with the execution, choose *Custom Info* from the *Metric Resource* tab options, or scroll down the screen.
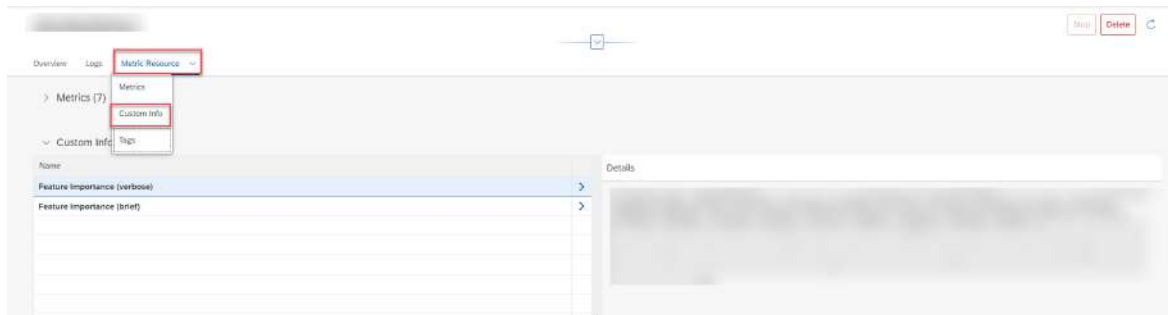
4. To display the tags associated with the execution, choose *Tags* from the *Metric Resource* tab options, or scroll down the screen.

Related Information

Compare Metric Resources [page 226]

6.2.4.4.9 Compare Executions

You can compare executions to determine which configuration parameters result in optimum results.

You can compare up to five executions, either by comparing metric data or by creating and analyzing charts.

- Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the execution (training process) and by the input configuration parameters.
  When you compare metrics for executions, multiple execution metrics are compared against input configuration parameters. You compare metric data to determine whether further adjustments are required for configurations, or to identify a configuration which is producing optimum results.
- Charts provide a visual representation of an execution and resulting model quality. You can choose the chart settings and chart type to graphically compare executions. Chart types include line, bar, column, heat map, and scattered charts. The chart types available for use depend on your source data and chart settings. You can preview a chart before you add it to your chart view.

> ⓘ **Note**
>
> You can only make comparisons if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

6.2.4.4.9.1 Create Chart to Compare Executions

You can create a chart to visually compare quality criteria and values for executions.

Context

You can create multiple charts and view them in the *Visual Board*.

> ⓘ **Note**
>
> Each chart can include data for up to five executions.

Procedure

1. In the *ML Operations* app, choose *Executions*.
2. **Optional:** If you have a large list of executions, you can filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.
    a. Enter the execution ID or select a status.
    b. Choose *Apply* to apply the filter to the list.
3. Select the executions for comparison and choose *View Metrics*.
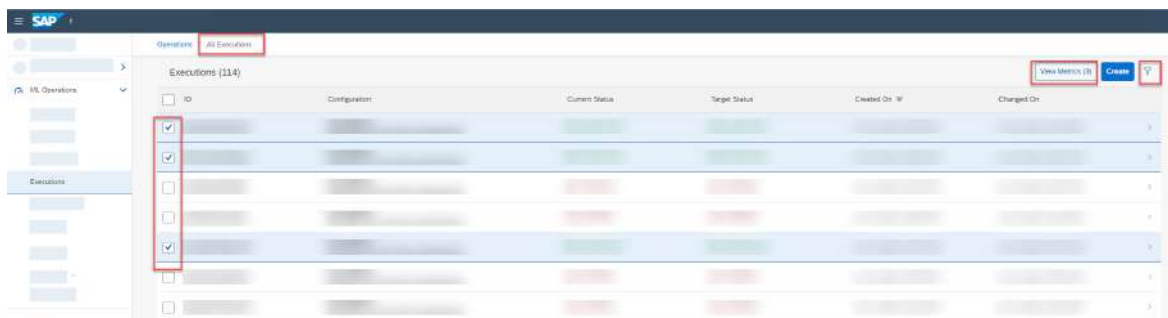
The *Metrics Overview* appears for the selected executions. The execution IDs and descriptions are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.
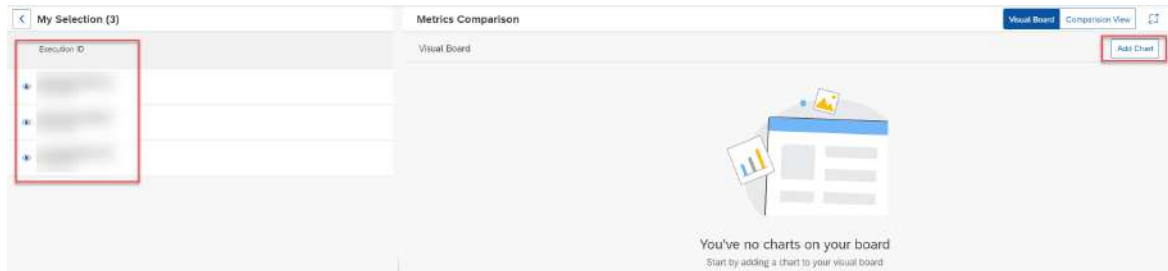
4. Choose *Add Chart* to create a chart based on your selected executions.

   The *Add Chart* dialog appears.

5. Enter the chart settings:

   - Enter a name and description for the chart.
   - In *Chart Settings*, choose `Executions` as the metrics source.
   - In *Comparison Type*, choose your preferred comparison. You can compare metrics to parameters, to the source, or to steps or time. Based on your selection, you'll be prompted to select the metrics and values for comparison.

6. Choose *Preview* to continue to the preview the chart settings.

   ⚠ Caution

   If *Preview* is not enabled, review your settings and selections. Some settings are mandatory, and you can't proceed until specified. Some settings and data combinations don't correspond to a valid chart type.

7. In the *Chart Selection* pane, select the chart type (such as column or bar chart). Note, the chart types available depend on the chart settings you defined.

8. Choose *Data Selection* menu option to confirm the executions selected for the chart.

   You can show or hide executions from your selection, and see the impact on the preview chart.
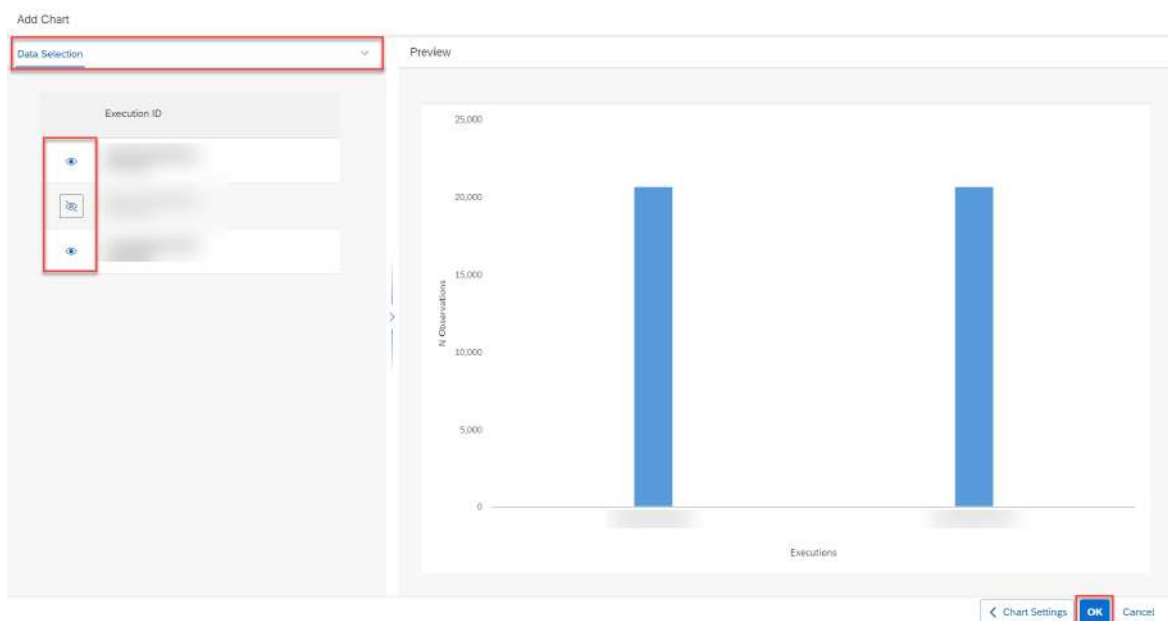
9. Choose *OK* to create the chart. The chart appears in your chart view.

10. **Optional:** Check the chart. Note, if you have multiple charts, you may need to scroll.
    a. To display a chart in full-screen mode, choose ⛶ *(Open Full Screen)*.
    b. To edit a chart, choose 🖉 *(Edit)*.
    c. To delete a chart from your visual board, choose 🖉 *(Delete)*.

6.2.4.4.9.2 Compare Metric Resources

You compare metrics resources for executions to determine which configuration parameters result in optimum results.

Context

Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the execution (training process).

When you compare metrics, multiple model/ execution metrics are compared against configuration parameters. You compare metric data to determine whether further adjustments are required, or to identify a configuration which is producing optimum results.

> ⓘ Note
>
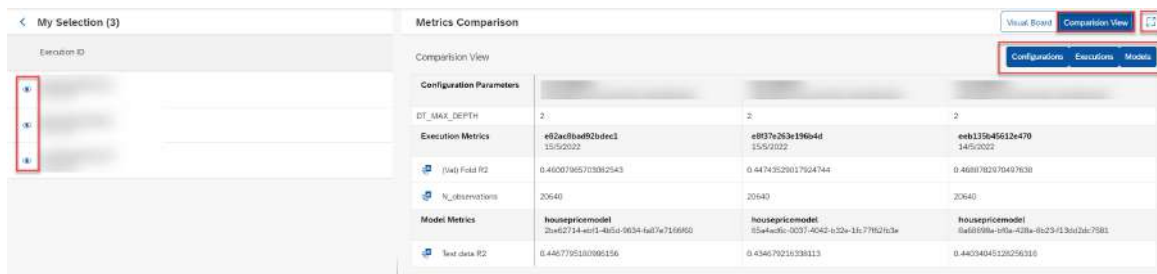> You can compare metrics for up to five executions.

Procedure

1. In the *ML Operations* app, choose *Executions*.
2. Select the executions for comparison and choose *View Metrics*.

   The *Metrics Overview* appears for the selected executions. The execution IDs and descriptions are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.
3. Choose *Comparison View* to compare metrics for your selected executions.
4. Investigate the metric details.

- To show detailed comparison results, choose ⤢ *(Expand)*.
- To show or hide metrics for an execution, choose ◉ *(Show)* or ◎ *(Hide)*.
- To show or hide different comparison criteria, choose *Configurations*, *Executions*, or *Models*.

6.2.4.4.10 View Execution Logs

You check the logs for an execution to debug an issue with a training pipeline.

Procedure

1. Navigate to the execution's details. See .
2. To view the execution logs, select the *Logs* tab on the execution details page.

   Log details include a timestamp and message. By default, logs are displayed in descending order based on timestamp.
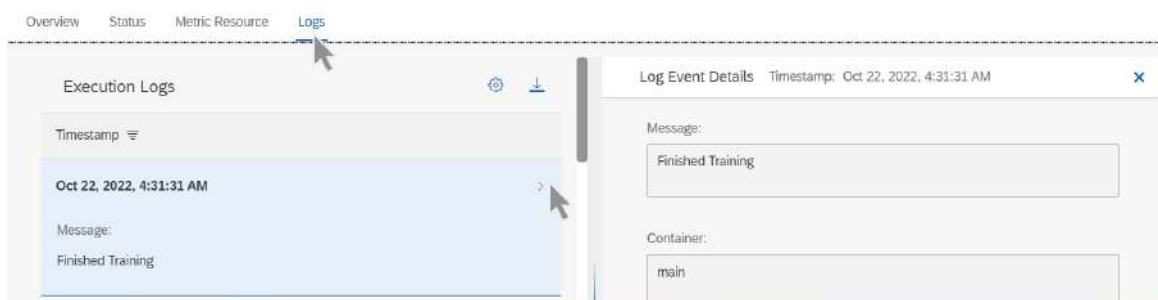
3. Choose ⚙ *(View Settings)* to sort ↓↑ *(Sort))* or filter ▽ *(Filter)* the logs by timestamp.
4. **Optional:** Choose ⤓ *(Download Logs)* to download the logs to your *Downloads* folder. You can download all logs or just those logs for a specified timestamp range.
5. To view the details for an individual log, select the log in the list or choose ❯ *(More)*. The log event details are displayed in the right pane.

6.2.4.5 Deployments

A deployment runs a model for serving (inferencing) purposes. You use deployments to make online predictions.

A serving executable is an AI pipeline that serves or deploys a model for online predictions. You use a configuration to specify values for the serving executable, such as the model as an input artifact. The configuration is then used to create a deployment.

A *running* deployment can only reference one configuration. However, you can create multiple configurations for use with a deployment. You can update a deployment whilst it is running to change the referenced configuration.

A deployment is an instance of a serving executable (referenced in a configuration). For more information, see Configurations [page 207]. Multiple deployments can be created using the same configuration, resulting in separate endpoints for online predictions.

You use SAP AI Launchpad to create deployments for your runtime connection. Deployments that are implemented on an SAP AI Core runtime produce `HTTPS` endpoints.

Custom Features

> → Remember
>
> The features that you see in the user interface depend on the capability settings in your underlying runtime. For more information, see Custom Runtime Capabilities Using the Meta API [page 43].

Deployment Duration

If your runtime supports deployment durations (such as SAP AI Core), you'll be able to define durations for your deployments. The duration you nominate (for example, two days) is added to the deployment's creation timestamp. The deployment runs until this time. Deployments which are not finished by the running until time are automatically stopped and deleted.

Deployment Quotas

Each tenant is assigned a default quota that limits the number of deployments and replicas per deployment. If you reach this quota, additional deployments can't be created, and you'll be notified. You can free up your quota by deleting deployments that are no longer required.

Deployment Statuses

Deployments can have any of the following statuses:

Pending
Running
Stopping
Stopped
Dead
Unknown

When a deployment has a *running* status, it can be used to make predictions.

> ⓘ Note
>
> A deployment with a *stopped* status cannot be restarted. To run a deployment again, create another deployment using the same configuration, and retain the combination of values for serving executable and model.

The following figure shows how deployment statuses change following the initial status of *Pending*:
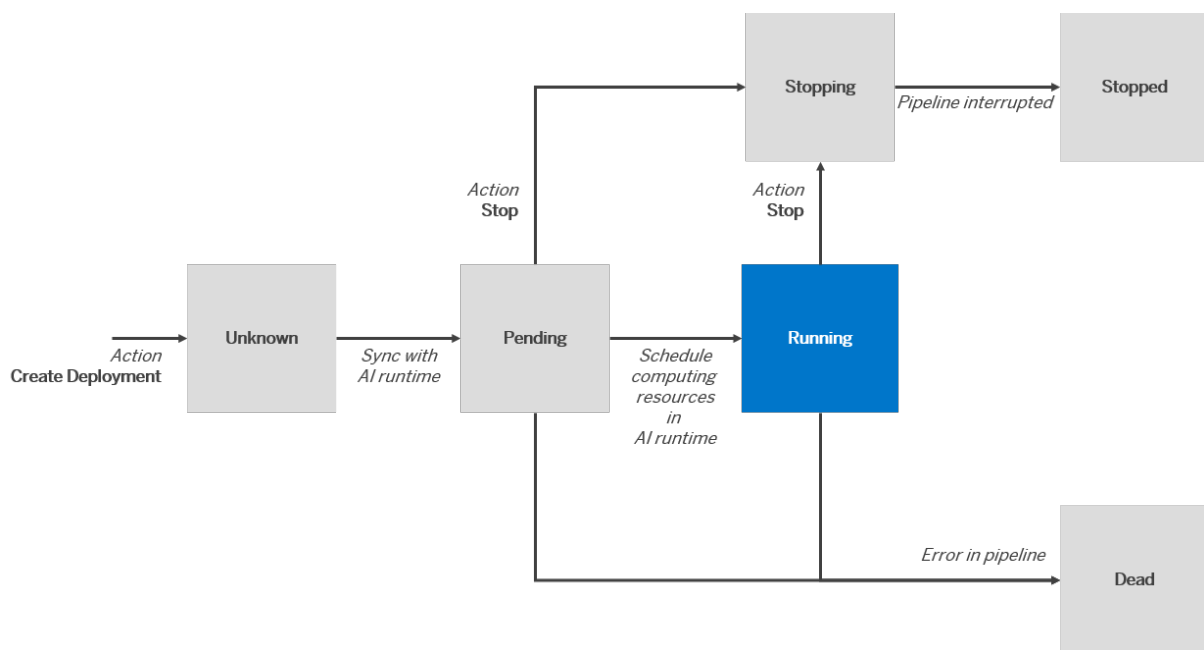
Deployment Status Flow

When a deployment leaves *Running* status, the computing resources that were used by the AI runtime are released.

Stop/ Delete Behavior by Deployment Status

Status	Stop Deployment	Delete Deployment
Unknown	Not enabled	Enabled

Status	Stop Deployment	Delete Deployment
Pending	Enabled	Not enabled
Running	Enabled	Not enabled
Stopping	Not enabled	Not enabled
Stopped	Not enabled	Enabled
Dead	Not enabled	Enabled

Related Information

Configurations [page 207]

6.2.4.5.1 Create a Deployment

You create a deployment to run a model for serving purposes.

Prerequisites

You have either the `mloperations_editor` or `scenario_deployment_editor` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

If your runtime supports deployment durations, you'll be able to set a timeframe for your deployment to run. The duration is determined by adding the specified timeframe to the creation timestamp. Deployments run until this time and are then automatically stopped and deleted.

Procedure

1. Select a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Deployments*.

The *All Deployments* screen appears listing all of the deployments for the selected resource group. Deployments are listed by ID, and with additional details such as configuration name and ID, current and target status, created on timestamp, and changed on timestamp.

3. Choose *Create Deployment* to create a new deployment.

   The *Create Deployment* wizard appears. This wizard has five steps.

4. Select the required data for the new deployment.

   1. In the *Select Scenario* step, select the scenario from the list and choose *Next*.
   2. In the *Select Executable* step, select the serving executable (deployment template) from the list and choose *Next*.
   3. In the *Select Configuration* step, select the required configuration. The details for the selected configuration are displayed in the right pane. You can search the list by entering a configuration name or part of the name in the Q *(Search)* field.

      → Tip

      When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

      → Tip

      If there is no configuration which matches your data requirements, you can choose *create a configuration*. You'll be redirected to create a configuration, and the deployment you've started will be lost. When you have saved the new configuration, you can re-create the deployment using the new configuration. See Create a Configuration.

   4. In the *Duration* step, select either the standard or custom duration.
      The standard duration defaults from your runtime. For example the standard duration in the SAP AI Core production environment is unlimited.
      If you choose a custom duration for the deployment, you can nominate a timeframe over minutes, hours, or days. When the running until time is reached, the deployment is stopped and deleted.

      → Tip

      Your runtime can define minimum and maximum duration, which can be used to limit duration. For more information on the duration defined for the SAP AI Core runtime, see AI API Runtime Capabilities Endpoint and Deploy Models.

      Choose *Review* to continue.

   5. In the *Review* step, review the data that you've selected for the new deployment.

      → Remember

      After you have created a deployment, you cannot change its duration.

      Choose *Create* to create the deployment.

Results

The new deployment is created and is now displayed on the *All Deployments* screen.

Related Information

6.2.4.5.2   View a Deployment

You can view the lifecycle details for a deployment, and explore details for each operation in the deployment.

Prerequisites

You have the `mloperations_viewer` or `scenario_deployment_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Deployments*.

   The *All Deployments* screen appears listing all of the deployments for the selected resource group. Deployments are listed by ID, and with additional details such as configuration name and ID, current and target status, created on timestamp, and changed on timestamp.

   > ⓘ Note
   >
   > If your runtime supports deployment durations, you'll see running until dates. Deployments run until this time, and are then automatically stopped and deleted.

3. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.
   a. Enter a deployment ID, if required. A valid deployment ID consists of lowercase characters, hyphens (-), and numbers.
   b. Alternatively, enter a configuration ID or status, or choose a scenario and additional attributes, such as version and executable.
   c. Choose *Apply* to apply the filter and conditions to the list.
4. To view the details for an individual deployment, select the deployment in the list or choose ❯ *(More)*. You'll see an overview of the deployment lifecycle process.
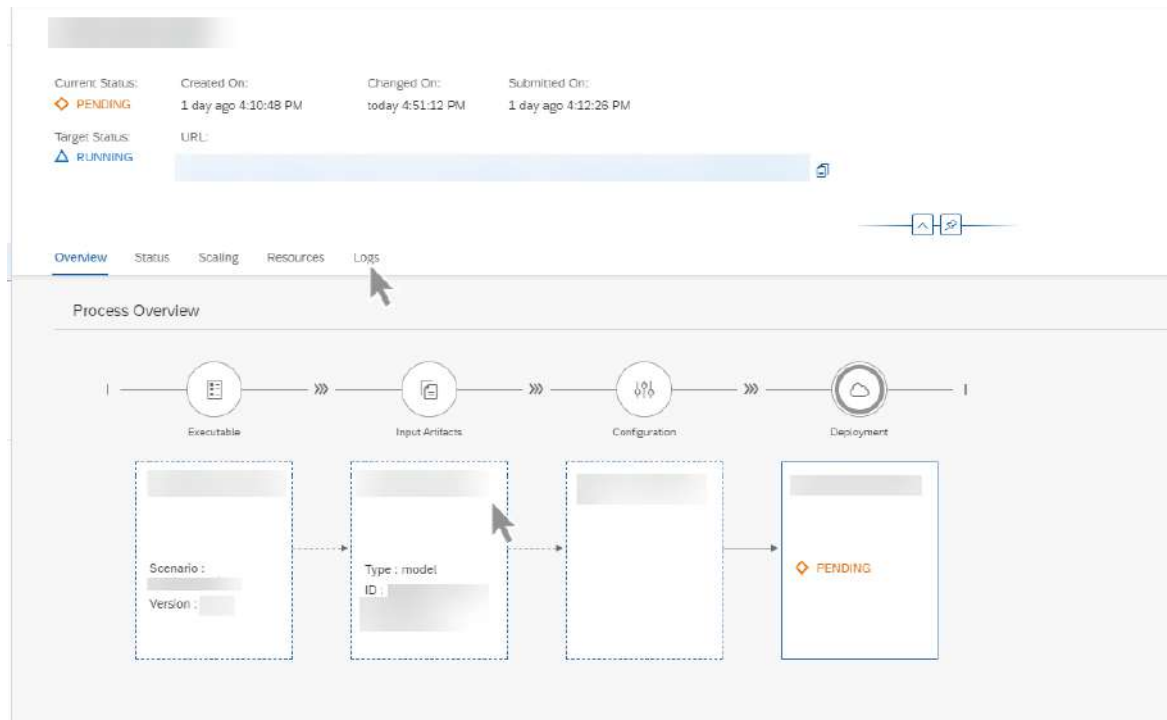
> → Tip
>
> Timestamp details for each step in the deployment process can be seen in the header. These dates and times show when the deployment was created, submitted, started, and finished. The process duration is also displayed. Timestamps are displayed in the user's local time zone.
>
> If your runtime supports deployment durations, you'll see a running until date. If a deployment is not complete by this time it will be automatically stopped and deleted.
>
> The *URL* is the endpoint used to make predictions.
>
> The *Overview* tab shows the operations involved in the deployment, as well as summary data, such as ID. You can click each card in the process overview to display the specific operation in more detail.

5. **Optional:** Choose the *Status* tab to check detailed message, status, and severity information for the deployment. See View Status Details.
6. **Optional:** When your runtime is SAP AI Core, you can choose the *Scaling* tab to check the replica limits, and the number of running replicas for the deployment.
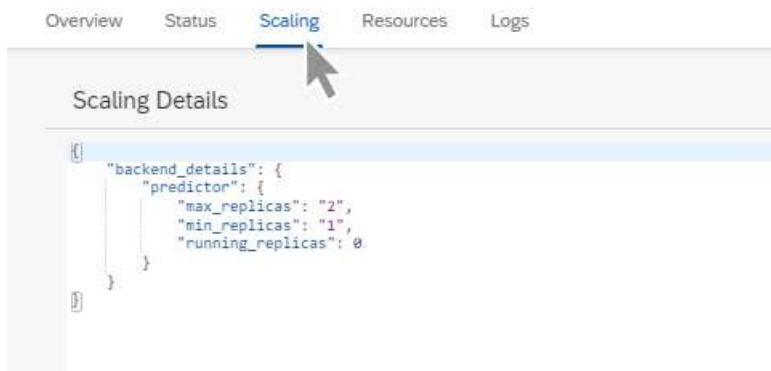
> → Tip
>
> A running deployment on a given node is called a replica. Replicas are used in distributed model training. For performance reasons, there are limits on the number of running replicas. You check how many replicas are running for a model deployment to better understand performance.

7. **Optional:** When your runtime is SAP AI Core, you can choose the *Resources* tab to check what resource plan applies to the deployment.

> → Tip
>
> A resource plan is a preconfigured infrastructure bundle that is used to process an AI operation, such as a deployment. Different operations within an AI workflow can use different resource plans.

8. **Optional:** Choose the *Logs* tab to check or download the logs for the deployment. See View Deployment Logs .

Related Information

Update a Deployment [page 235]

6.2.4.5.3   Update a Deployment

You can update a deployment with your choice of configuration.

Prerequisites

You have the `mloperations_editor` role, or you are assigned a role collection that contains this role. For more information, see Roles and Authorizations.

Context

Multiple configurations can be associated with a deployment. However, a deployment can only use one configuration. When you update a deployment, you can choose to apply a new configuration, or re-use an older or previously used configuration. For example:

- Your existing deployment serves models A and B, but now you'd like to replace model A with a new model C. You want to use the same deployment and perform inference using the existing deployment URL endpoint.
- You want to update an existing deployment with new parameter bindings, such as number of replicas, autoscaling annotation, or ENV variables.

Deployments can only be updated if their status is *pending*, *running* or *dead*.

For deployments subject to a duration, there is no impact on duration when updating the deployment. The deployment will run with the updated configuration until the specified date and time.

The updated deployment retains the inference URL. If you use a new deployment configuration, inference requests continue to work.

Procedure

1. Navigate to the deployment's details. See View a Deployment.
2. Choose *Update*. The *Update Deployment* dialog box appears listing all available configurations. You can display summary details for each configuration, and use this information to compare details such as parameters and input artifacts.
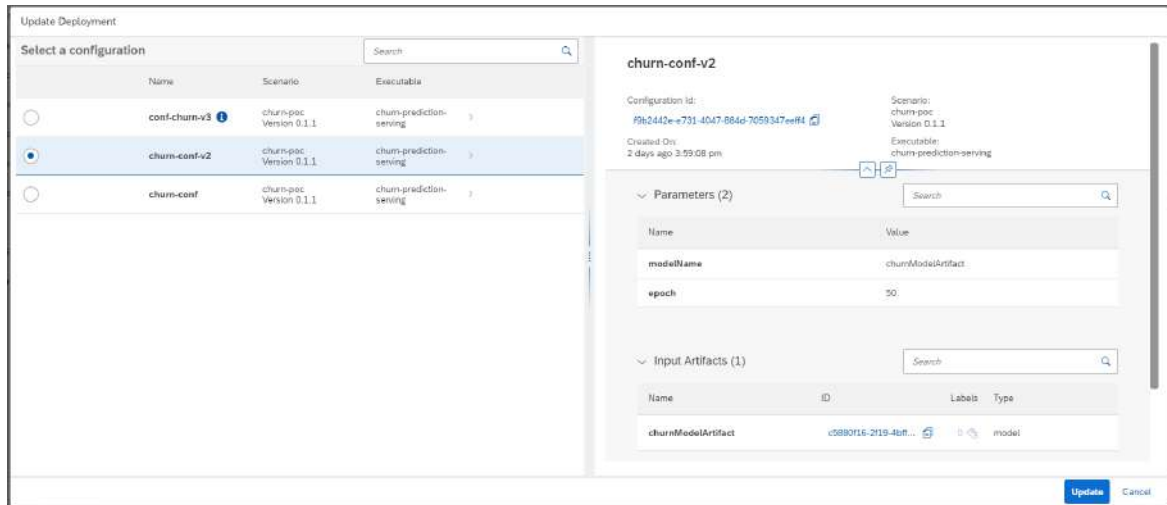
3. Select the required configuration from the list, and choose *Update* to update the deployment.

> → Tip
>
> The system shows the configuration which is already in use. You can't update the deployment with this configuration.

4. Check the *Overview* tab of the *Deployment Details* page, and confirm that the deployment is *running* with the selected configuration.

> ⓘ Note
>
> If the update was not effective and the deployment status is *dead*, you'll receive an error message. The error message contains the configuration ID of the last configuration that resulted in a *running* deployment. You can update the deployment with this last configuration to undo or rollback the error, and return the deployment to a *running* status.
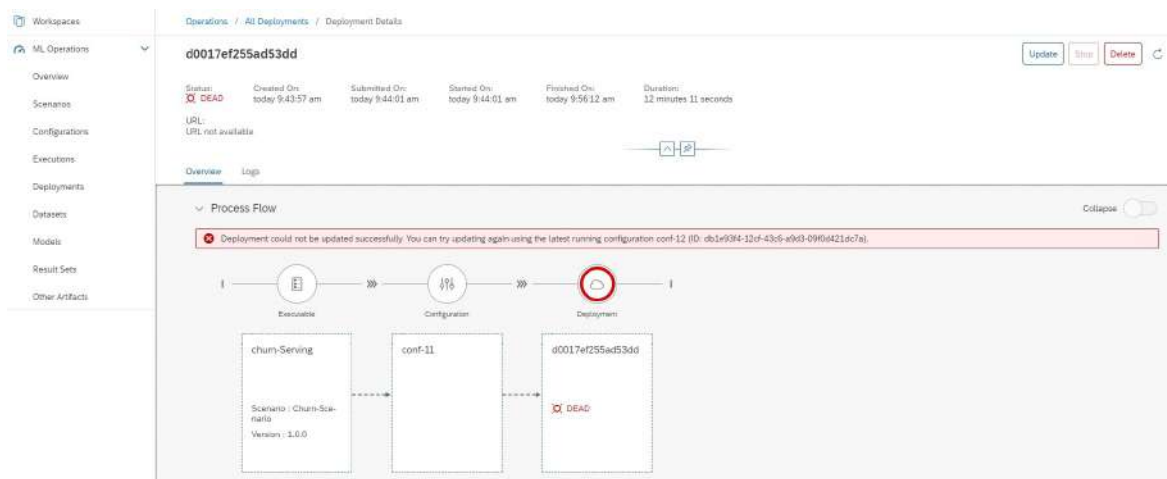
6.2.4.5.4 View Status Details

You check a deployment's status details for detailed message, status, and severity information for a running deployment.

Prerequisites

You have the `mloperations_viewer` or `scenario_deployment_viewer` role, or you are assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

Once a deployment is created, the status details provide information about a running deployment.

Procedure

1. Navigate to the deployment's details. See View a Deployment.
2. Choose the *Status* tab to view status details. You may need to scroll to check all status details for the deployment.

6.2.4.5.5 Stop a Deployment

Stopping a deployment releases the computing resources acquired in the runtime in which the deployment is present (such as SAP AI Core).

Prerequisites

You have the `mloperations_editor` or `scenario_deployment_editor` role, or you are assigned a role collection that contains this role. For more information, see Roles and Authorizations.

Context

You can stop a deployment if it has a *Running* or *Pending* status. For deployments in other statuses, the *Stop* button is not enabled.

Procedure

1. Navigate to the deployment's details. See View a Deployment.
2. Choose *Stop* in the header. A *Warning* dialog box appears.
3. Choose *Stop* to stop running the deployment.

Results

The deployment is stopped, and the computing resources that your deployment was using in the runtime instance are released.

6.2.4.5.6 Stop Multiple Deployments

Prerequisites

You have the `mloperations_editor` or `scenario_deployment_editor` role, or you are assigned a role collection that contains this role. For more information, see Roles and Authorizations.

Context

- The maximum number of updates per request is 100.
- Only *stopped*, *dead* or *unknown* executions or deployments can be deleted.
- Only *running* or *pending* executions or deployments can be stopped.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Deployments*.

   The *Deployments* screen appears listing all of the deployments for the selected resource group. Deployments are listed by ID, and with additional details such as Name, Current and Target Status, Created On timestamp, and Changed On timestamp.
3. Choose the Deployments you want to stop, using the checkboxes.
4. Choose *Stop* in the header.

Results

A dialog box appears, confirming how many of your selected deployments will be stopped.

> ⓘ Note
>
> Where executions or deployments with a mixture of states that can be stopped **and** deleted are selected, only items eligible to the chosen request will be successfully processed.

6.2.4.5.7   Delete a Deployment

You delete a deployment to remove it from your instance.

Prerequisites

You have either the `mloperations_editor` or `scenario_deployment_editor` role, or you are assigned to a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

Deployments are subject to quota limits. If you reach your quota limit, additional deployments can't be created and you'll be notified. You can free up your quota by deleting deployments that are no longer required.

You can delete a deployment if it has the status *Dead*, *Stopped*, or *Unknown*. For deployments in other statuses, the *Delete* button is not enabled.

Procedure

1. Navigate to the deployment's details. See View a Deployment.
2. Choose *Delete* in the header. A *Warning* dialog box appears.
3. Choose *Delete* to confirm the deletion.

Results

The deployment is deleted, and its associated information is removed and cannot be retrieved.

> ⓘ Note
>
> To create a deployment with the same settings (model) as the deleted deployment, you can use the same configuration to create a new deployment. The new deployment will have the same settings, however, it will generate a different endpoint (URL).

6.2.4.5.8 Delete Multiple Deployments

Prerequisites

You have the `mloperations_editor` or `scenario_deployment_editor` role, or you are assigned a role collection that contains this role. For more information, see Roles and Authorizations.

Context

- The maximum number of updates per request is 100.
- Only *stopped*, *dead* or *unknown* executions or deployments can be deleted.

- Only *running* or *pending* executions or deployments can be stopped.

Procedure

1. Choose a resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Deployments*.

   The *Deployments* screen appears listing all of the deployments for the selected resource group. Deployments are listed by ID, and with additional details such as Name, Current and Target Status, Created On timestamp, and Changed On timestamp.
3. Choose the Deployments you want to stop, using the checkboxes.
4. Choose *Delete* in the header.

Results

A dialog box appears, confirming how many of your selected deployments will be deleted.

> ⓘ Note
>
> Where executions or deployments with a mixture of states that can be stopped **and** deleted are selected, only items eligible to the chosen request will be successfully processed.

6.2.4.5.9   View Deployment Logs

Deployment logs are generated by the code (in the AI pipeline template) which is deploying the model.

Context

Deployment logs differ from deployment status logs (View Status Details).

- Status logs show information **about** the computing resources and components (models and AI pipeline template)
- Deployment logs show information **from** the computing resources and components

Procedure

1. Navigate to the deployment's details screen. See View a Deployment.

2. Choose the *Logs* tab to display log details.

   Log details include a timestamp and message. By default, logs are displayed in descending order based on timestamp.
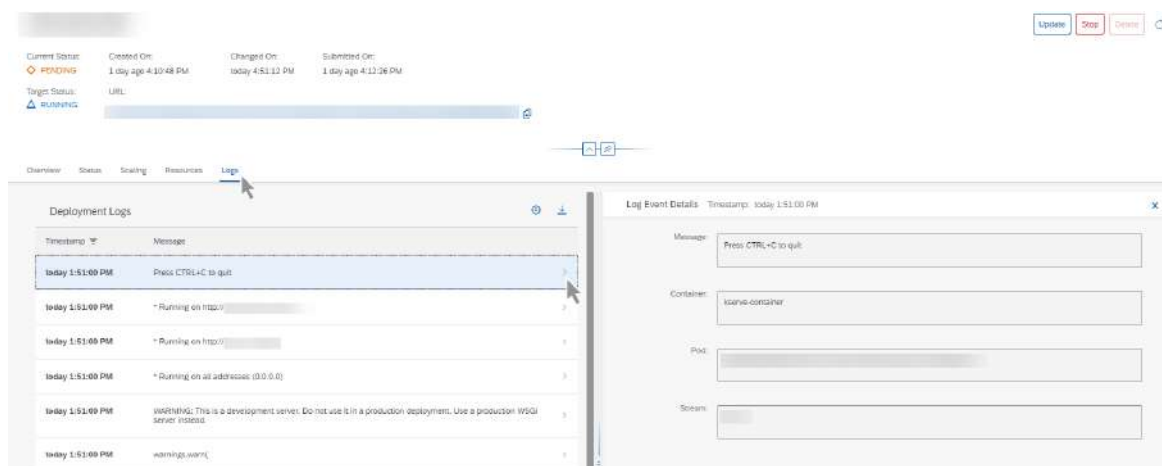
3. Choose ⚙ *(View Settings)* to sort ↓↑ *(Sort))* or filter ▽ *(Filter)* the logs by timestamp.

4. **Optional:** Choose ⬇ *(Download Logs)* to download the logs to your *Downloads* folder. You can download all logs or just those logs for a specified timestamp range.

5. To view the details for an individual log, select the log in the list or choose ❯ *(More)*. The log event details are displayed in the right pane.

6.2.4.6 Schedules

Set up a periodic scheduling of workflow executions to automate execution creation, and define its parameters such as the recurrence period and recurrence duration. Users can update a schedule to change its recurrence period of operation and job.

Custom Features

> → Remember
>
> The features that you see in the user interface depend on the capability settings in your underlying runtime or service. For more information, see Custom Runtime Capabilities Using the Meta API [page 43].

Schedule Statuses

The schedule statuses are as follows:

- **Active:** The schedule created is currently running and will create executions based on the specified periodicity.

- **Inactive:** The schedule is either manually stopped or it has reached its end time. It will no longer create executions.

6.2.4.6.1   Create a Schedule

Prerequisites

You have the `mloperations_editor` or `execution_schedules_editor role`, or you have been assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Choose a resource group. For more information, see Set Resource Group [page 164].
2. In the *ML Operations* app, choose *Schedules*.

   The **All Schedules** screen appears listing all of the schedules for the selected resource group.
3. Choose *Create* to create a new schedule.
   The *Create Schedule* wizard appears. This wizard has five steps.
4. Select the required data for the new deployment.

   1. In the *Select Scenario* step, select the scenario from the list and choose *Next*.
   2. In the *Select Executable* step, select the workflow executable from the list and choose *Next*.
   3. In the *Select Configuration* step, select the required configuration. The details for the selected configuration are displayed in the right pane. You can search the list by entering a configuration name or part of the name in the Q *(Search)* field.

      > → Tip
      >
      > If there is no configuration which matches your data requirements, you can choose *create a configuration*. You'll be redirected to create a configuration, and the schedule you've started will be lost. When you have saved the new configuration, you can re-create the schedule using the new configuration. For more information, see Create a Configuration [page 209].

5. In the *Schedule* step, choose from:

   - To create a *One-Time* schedule, choose *One-Time* schedule and selecting the end date and time.

- To create a *Recurring* schedule choose *Recurring*. Optionally, you can enter a cron expression to provide a start and end time.

> ⓘ Note
>
> Your AI runtime can specify a minimum frequency expected for the cron expression. For more information, see About the AI API.

> ⓘ Note
>
> The cron expression entered is in UTC time. For more information, see Cron Formatting ↗ .

6. Choose *Review* to continue.

6.2.4.6.2 View a Schedule

Prerequisites

You have the `mloperations_viewer` or `execution_schedules_viewer` role, or you have been assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations [page 321].

Procedure

1. Choose a resource group. For more information, see Set Resource Group [page 164].
2. In the *ML Operations* app, choose *Schedules*.

   The *Schedules* screen appears listing all of the schedules for the selected resource group. Schedules are listed by name/ID, with additional details such as Configuration ID, Status, and Created On timestamp. Details for the selected schedule are displayed on the right.

3. Filter the list by choosing ∇ *(Filter)*. The *Filter* dialog appears. Enter at least one of the following:

   - Schedule ID: A valid schedule ID consists of lowercase characters, hyphens (-), and numbers
   - Configuration ID
   - Status

   Choose *Apply* to apply the filter and conditions to the list.

4. When a schedule is selected, you can see the details of the selected schedule in the adjacent panel.

   - For **One-time** schedules, the *Schedule Details* tab shows the creation time stamp.
   - For **Recurring** schedules, the *Schedule Details* tab shows the cron expression with the start and end time of the schedule.

   > ⓘ Note
   >
   > The cron expression entered is in UTC time.
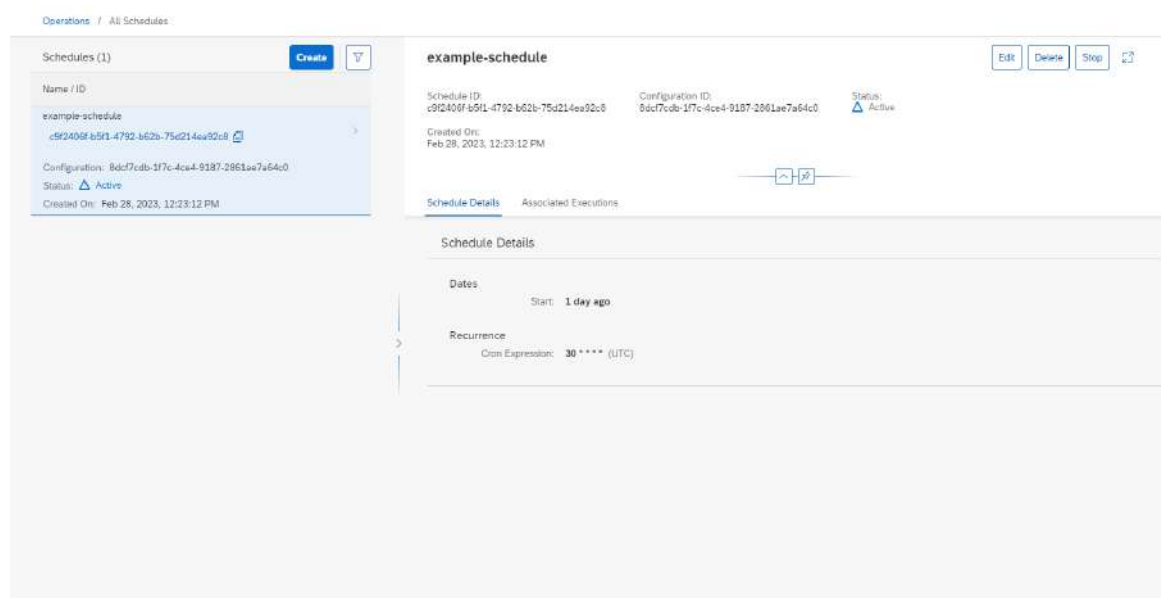
5. Choose the *Associated Executions* tab to view the executions created by the schedule. For more information, see View Associated Executions [page 245]
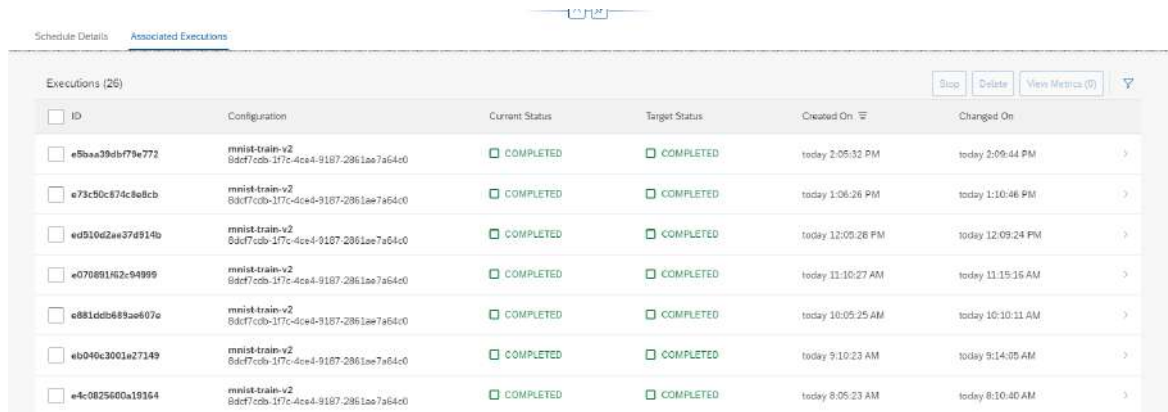
6.2.4.6.3 View Associated Executions

Context

You have the `mloperations_editor` or `execution_schedules_viewer` role, or you have been assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations [page 321].

Executions created by this schedule are shown in the *Associated Executions* tab.

Procedure

1. Navigate to the Schedule's details. For more information, see View a Schedule [page 244].
2. Choose the *Associated Executions* tab to view the executions created by this schedule.

6.2.4.6.4   Stop or Resume a Schedule

Prerequisites

You have the `mloperations_editor` or `execution_schedules_editor` role, or you have been assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

You can only stop an active schedule or resume an inactive schedule if its end time has not been reached.

Procedure

1. Navigate to the Schedule's details. For more information, see View a Schedule [page 244].
2. Choose from:

   - *Stop* to stop an active schedule.
   - *Resume* to resume an inactive schedule.

   The schedule status will change.

6.2.4.6.5 Edit a Schedule

Edit a schedule to change its recurrence, period of operation and the configuration used to create executions.

Prerequisites

You have the `mloperations_editor` or `execution_schedules_editor role`, or you have been assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Procedure

1. Navigate to the Schedule's details. For more information, see View a Schedule [page 244].
2. Choose *Edit*.

   The delete schedule dialog box appears, with option to *Review* your steps.
3. Navigate to the relevant step and make your changes.

4. Confirm your changes on the *Review* page.

5. Press *Edit* to save your edits.

6.2.4.6.6   Delete a Schedule

Prerequisites

You have the `mloperations_editor` or `execution_schedules_editor` role, or you have been assigned a role collection that contains one of these roles. For more information, see Roles and Authorizations.

Context

Deleting a schedule only deletes the schedule itself. It does not delete executions which were created by the schedule.

Procedure

1. Navigate to the Schedule's details. For more information, see View a Schedule [page 244].
2. Choose *Delete*.

   The delete schedule dialog box appears.
3. Choose *Delete*.

   The schedule will be deleted.

   > ⓘ Note
   >
   > Deleting a schedule does not delete the executions created by the schedule.

Related Information

Delete an Execution [page 220]

6.2.4.7   Datasets

A dataset is a type of artifact which is registered in your AI runtime. A registered dataset references files that are stored in your connected hyperscaler object store.

Data is uploaded to your hyperscaler object store as a single file (such as a `CSV` file) or as multiple files (such as images). The data is then registered as a dataset with a unique ID. You can use the dataset ID in a configuration as an input artifact value in a workflow executable.

SAP AI Launchpad can be used with multiple hyperscaler object stores, such as Amazon S3, OSS, and WebHDFS.

Datasets are unique to a resource group.

About Dataset Registration

- A dataset can be manually registered using SAP AI Launchpad (see Register a Dataset [page 250]). You register a dataset using the *ML Operations* app. After you register a dataset, you'll see it listed with all other datasets in the app.
- A dataset can be manually registered using SAP AI Core (see Create Artifacts).

Related Information

Artifact Management

6.2.4.7.1 Register a Dataset

Use the *ML Operations* app to manually register a dataset that is stored in your object store.

Prerequisites

You have the `artifact.register` role, or you have been assigned a role collection that contains this role. For more information, see Roles and Authorizations.

Details for the object store secret have been added using the *SAP AI Core Administration* app. See Add an Object Store Secret.

Context

After you've registered a dataset, you can use it as input for configurations in the *ML Operations* app.

> ⚠ Caution
>
> If the files for a registered dataset are deleted from your object store, or if the datapath or object store secret changes, then the registered dataset can no longer be used.

Procedure

1. Choose the resource group. For more information, see Set Resource Group.
2. In the *ML Operations* app, choose *Datasets*.

   The *Datasets* screen appears listing all of the datasets for the selected resource group.
3. Choose *Add* to manually register a dataset.

   The register dataset wizard appears. This wizard has five steps.
4. Enter the details for the dataset.

   a. In the *Scenario* step, choose the scenario in which the dataset will be used.
   b. In the *General Information* step, specify a name and description for the dataset.
   c. In the *URL* step, enter the URL for the dataset in your object store.

      > → Tip
      >
      > URLs must adhere to the following convention `ai://<objectStore name>/<data path>`.

   d. In the *Labels* (Optional) step, enter any labels and values that you would like to apply to the dataset.
   e. In the *Review* step, confirm the details that you've provided for the dataset.
5. Choose *Add* to complete dataset registration.

Results

The dataset that you have registered now appears in the list of datasets for the resource group.

6.2.4.7.2 Find a Dataset

Use the *ML Operations* app to search for a dataset.

Prerequisites

- You have files which are stored in hyperscaler object storage and which have been registered as a dataset.
- You have the `scenario_artifact_viewer` role or you are assigned a role collection that contains it. For more information, see Roles and Authorizations.
- You have specified the resource group as described at Set Resource Group.

Procedure

1. In the *ML Operations* app, choose *Datasets*.

   The *Datasets* screen appears listing all of the datasets registered to your resource group for your runtime.

   Datasets are listed by name, and with additional details such as description, ID, scenario, URL, execution ID, labels, created on timestamp, and changed on timestamp.

2. **Optional:** Search the list by entering a dataset name or part of the dataset description in the 🔍 *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

3. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

a. Enter the dataset name or other attributes.

b. Choose *Add* to include the filter criteria. Adjust the conditions as needed. You can add up to 10 filter conditions.

c. Choose *Apply* to apply the filter and conditions to the list.

Once you've found the dataset, you can note the ID for use in a configuration.

6.2.4.8 Models

A model is a type of artifact that results from a training process.

Models are stored as files in your connected hyperscaler object storage. The object storage is associated with your AI runtime connection.

SAP AI Launchpad can be used with multiple hyperscaler object stores, such as Amazon S3, OSS, and WebHDFS.

Models are unique to a resource group.

You can create a model, by running an execution in the *ML Operations* app. Models which are created this way, also include details of the source execution. Alternatively, you can reference an existing model by registering it for use.

About Model Registration

- Model creation: A model is created as the output of an execution within the *ML Operations* app. Models which result from this training process are **automatically** stored as files in your object store, and are automatically registered as model artifacts. Each model is assigned a unique model ID.

- Model registration: A model is manually registered using SAP AI Launchpad (see Register a Model [page 255]). Registering a model enables you to reference an existing model which is stored in hyperscaler data storage. After you register a model, you'll see it listed with all other models in the app. You register a model using the *ML Operations* app.

- Model registration: A model is manually registered using SAP AI Core (see Create Artifacts).

Related Information

Artifact Management

6.2.4.8.1 Investigate a Model

Use the *ML Operations* app to list the models for your selected connection, and investigate their origin.

Prerequisites

You have the `scenario_artifact_viewer` or `scenario_metric_viewer` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A model consists of both overview and metric data. Overview data summarizes the labels, parameters, and inputs associated with the source execution. Metric data is affected by the dataset and parameter values used by the execution, and can be used to determine a model's quality.

You can create a chart to visualize the performance of an individual model, or create a chart to compare models.
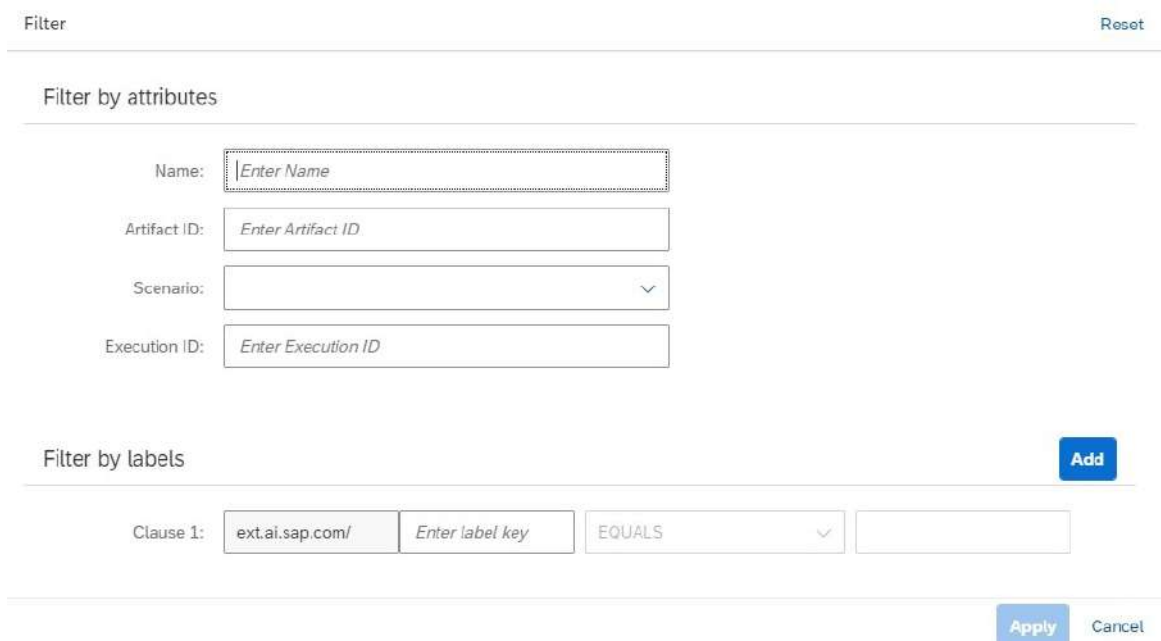
Procedure

1. Choose the resource group. For more information, see Set Resource Group [page 164].
2. In the *ML Operations* app, choose *Models*.

   The *Models* screen appears listing all of the models for the selected resource group. Models are listed by name, and with additional details such as description, ID, scenario, URL, execution ID, labels, created on timestamp, and changed on timestamp.

   > → Remember
   >
   > You use a model ID to uniquely identify a model. This ID is used in your configuration as a value for a placeholder (input artifact) of a serving executable (a pipeline to deploy an AI model).

3. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

Filter				Reset
Filter by attributes				
Name:	Enter Name			
Artifact ID:	Enter Artifact ID			
Scenario:			⌄	
Execution ID:	Enter Execution ID			
Filter by labels				**Add**
Clause 1:	ext.ai.sap.com/	Enter label key	EQUALS ⌄	
			Apply	Cancel

   a. Enter the model name or other attributes.
   b. Choose *Add* to include the filter criteria. Adjust the conditions as needed.
   c. Choose *Apply* to apply the filter and conditions to the list.

4. **Optional:** Search the list by entering a model name or part of the model's description in the 🔍 *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

5. Choose 🗐 *(Copy)* for your model to copy the model's unique ID.

   > → Tip
   >
   > To see how a model ID is used during a deployment, see Create a Deployment [page 230].

6. **Optional:** To create a chart for a model, select a model in the list and choose *View Metrics*. See Create Chart to Compare Models [page 258].

7. To view the details for a model and investigate its origin information, select a model in the list or choose ❯ *(More)*.

8. **Optional:** For data about the quality (confidence) of a model, see View Metrics for a Model [page 256].

6.2.4.8.2 Register a Model

Use the *ML Operations* app to manually register a model that is stored in your object store.

Prerequisites

You have the `artifact.register` role, or you have been assigned a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

For more information, see Roles and Authorizations [page 321].

Details for the object store secret have been added using the *SAP AI Core Administration* app. See Add an Object Store Secret [page 173].

Context

After you've registered a model, you can use it with deployments in the *ML Operations* app.

> ⚠ Caution
>
> If the files for a registered model are deleted from your object store, or if the datapath or object store secret changes, then the registered model can no longer be used.

Procedure

1. Choose the resource group. For more information, see Set Resource Group [page 164].
2. In the *ML Operations* app, choose *Models*.

   The *Models* screen appears listing all of the models for the selected resource group.
3. Choose *Add* to manually register a model.

   The register model wizard appears. This wizard has five steps.
4. Enter the details for the model.
   a. In the *Scenario* step, choose the scenario in which the model will be used.
   b. In the *General Information* step, specify a name and description for the model.
   c. In the *URL* step, enter the URL for the model in your object store.

      > → Tip
      >
      > URLs must adhere to the following convention `ai://<objectStore name>/<data path>`.

   d. In the *Labels* (Optional) step, enter any labels and values that you would like to apply to the model.
   e. In the *Review* step, confirm the details that you've provided for the model.

5. Choose *Add* to complete model registration.

Results

The model's details screen appears. Unlike models that result from an execution, registered models don't contain source execution details. The model that you have registered is now available for use, and is listed with all the models for the resource group.

6.2.4.8.3   View Metrics for a Model

The *Metrics* tab provides an overview of the quality of a model. The metric data is logged by a workflow executable during an execution (training process).

Prerequisites

You have the `scenario_artifact_viewer` or `scenario_metric_viewer` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the execution (training process), which you can check in Investigate a Model [page 253]. Metric data can help you to decide whether a model is performing well, and whether it is fit for purpose.

Metric data is common to both the generated models and executions. You can view the same metrics information on the execution's details page. For more information, see View the Metric Resource for an Execution [page 222].

Procedure

1. In the *ML Operations* app, find the model and display its details see Investigate a Model [page 253].
2. To view the model metrics, select the *Metrics* tab.

> ⓘ Note
>
> The *Metrics* tab is only visible if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

Metrics details are displayed, as follows:

- *Name*: Quality criteria (model evaluation metrics). For example, accuracy or mean absolute error (MSE).
- *Value*: Indicator of quality, which is dependent on the *Name* (criteria).
- *Timestamp* and *Step*: Used to uniquely identify or differentiate the results. For example, a model can train iteratively using the same dataset in a single training process. Also known as an epoch.
- *Labels*: Classifying phrase/ name applied to the metric for that training (workflow executable).

For a detailed comparison of metrics and their performance, see Compare Model Metrics [page 259].

To create a chart to compare models, see Create Chart to Compare Models [page 258].

Related Information

Querying Metric Data
Storing Metric Data
Compare Model Metrics [page 259]

6.2.4.8.4   Compare Models

You can compare models to determine which configuration parameters result in optimum results.

You can compare up to five models, either by comparing metric data or by creating charts.

- Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the execution (training process) and by the input configuration parameters.
  When you compare metrics for executions, multiple execution metrics are compared against input configuration parameters. You compare metric data to determine whether further adjustments are required for configurations, or to identify a configuration which is producing optimum results.
- Charts provide a visual representation of a run and resulting model quality. You can choose the chart settings and chart type to graphically compare models. Chart types include line, bar, column, heat map, and scattered charts. The chart types available for use depend on your source data and chart settings. You can preview a chart before you add it to your chart view.

> ⓘ Note
>
> You can only make comparisons if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

6.2.4.8.4.1 Create Chart to Compare Models

You can create charts for models to visually compare quality criteria and values.

Context

You create a chart to visualize metrics for:

- A single model, or
- Multiple models, comparing up to five models against timestamp, step, or configuration parameters

You can create multiple charts and view them in the *Visual Board*.

> ⓘ Note
>
> Each chart can compare data for up to five models.

Procedure

1. In the *ML Operations* app, choose *Models*.

   The *All Models* screen appears listing all of the models for the selected resource group. Models are listed by name, and with additional details such as description, ID, scenario, URL, execution ID, labels, created on timestamp, and changed on timestamp.

2. **Optional:** If you have a large list of models, you can further filter the list by choosing ▽ *(Filter)*.

3. **Optional:** Search the list by entering a model name or part of the model's description in the ⌕ *(Search)* field.

4. Select the models for comparison and choose *View Metrics*.

   The *Metrics Overview* appears for the selected models. The model IDs and descriptions are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.
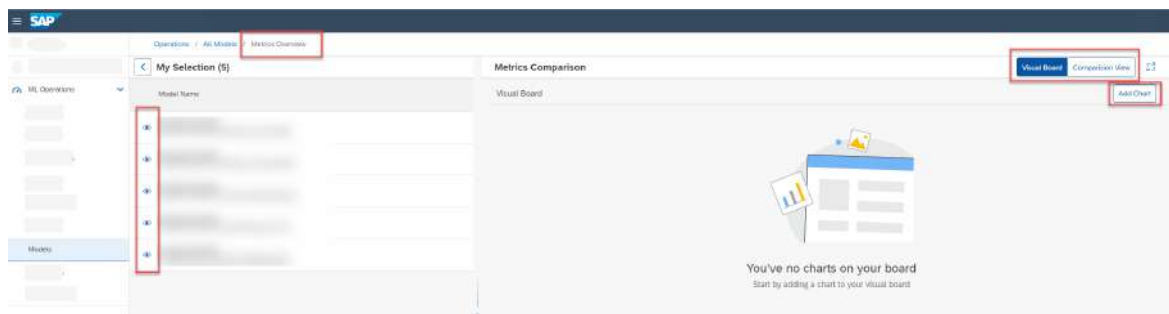
5. Choose *Add Chart* to create a chart comparing the selected models.

   The *Add Chart* dialog appears.

6. Enter the chart settings:

- Enter a name and description for the chart.
- In *Chart Settings*, choose `Models` as the metrics source.
- In *Comparison Type*, choose your preferred comparison. You can compare metrics to parameters, to the source, or to steps or time. Based on your selection, you'll be prompted to select the metrics and values for comparison.

7. Choose *Preview* to continue to the preview settings.

> ⚠ Caution
>
> If *Preview* is not enabled, review your settings and selections. Some settings are mandatory, and you can't proceed until specified. Some settings and data combinations don't correspond to a valid chart type.

8. In the *Chart Selection* pane, select the chart type (such as column or bar chart). Note, the chart types available depend on the chart settings you defined.

9. Choose *Data Selection* from the menu option to confirm the models selected for the chart.

   You can show or hide models from your selection, and see the impact on the preview chart.

10. Choose *OK* to create the chart. The chart appears in your chart view.

11. **Optional:** Check the chart. Note, if you already have multiple charts on your visual board, you may need to scroll.

   a. To display a chart in full-screen mode, choose ⤢ *(Open Full Screen)*.
   b. To edit a chart, choose ✎ *(Edit)*.
   c. To delete a chart from your chart view choose ✎ *(Delete)*.

6.2.4.8.4.2 Compare Model Metrics

You can compare metrics for models to determine which configuration parameters result in optimum results.

Context

Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the model (training process).

When you compare metrics, multiple model/ execution metrics are compared against configuration parameters. You compare metric data to determine whether further adjustments are required, or to identify a configuration which is producing optimum results.

> ⓘ Note
>
> You can compare metrics for up to five models.

Procedure

1. In the *ML Operations* app, choose *Models*.
2. Select the models for comparison and choose *View Metrics*.

    The *Metrics Comparison View* appears for the selected models.
3. Investigate the metric details.

    - To show detailed comparison results, choose ⤢ *(Expand)*.
    - To show or hide metrics for a model, choose ⊙ *(Show)* or ⦸ *(Hide)*.
    - To show or hide different comparison data, choose *Configurations*, *Executions*, or *Models*.

6.2.4.9 Result Sets

A result set is a type of artifact that results from an execution used for batch inferencing.

Result sets are registered as artifacts in the connection where your executions run. Result sets are generated from the workflow executable that is used in your execution process. The execution is not used for a training process, but instead used for batch inferencing and to make predictions on sequential subsets of data.

Result sets are stored as artifacts in your connected hyperscaler object storage. SAP AI Launchpad can be used with multiple hyperscaler object stores, such as Amazon S3, OSS, and WebHDFS.

Result sets are unique to a resource group.

> ⓘ Note
>
> Workflow templates in SAP AI Launchpad are available based on the selected connection to an AI runtime. For more information, see Workflow Templates. For information about how to create a workflow executable and include it in your SAP AI Core instance, see Setting Up Your Git Repository.

About Result Set Registration

- A result set can be manually registered using SAP AI Launchpad (see Register a Result Set [page 261]). You register a result set using the *ML Operations* app. After you register a result set, you'll see it listed with all other result sets in the app.
- A result set can be manually registered using SAP AI Core (see Create Artifacts).

Related Information

Artifact Management

6.2.4.9.1 Register a Result Set

Use the *ML Operations* app to manually register a result set that is stored in your object store.

Prerequisites

You have the `scenario_artifact_viewer` or `scenario_metric_viewer` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Details for the object store secret have been added using the *SAP AI Core Administration* app. See Add an Object Store Secret [page 173].

Context

After you've registered a result set, you can use it in the *ML Operations* app.

> ⚠ Caution
> If the files for a registered result set are deleted from your object store, or if the datapath or object store secret changes, then the registered result set can no longer be used.

Procedure

1. Choose the resource group. For more information, see Set Resource Group [page 164].
2. In the *ML Operations* app, choose *Result Sets*.

The *Result Sets* screen appears listing all of the results sets for the selected resource group.

3. Choose *Add* to manually register a result set.

   The register result set wizard appears. This wizard has five steps.

4. Enter the details for the result set.

   a. In the *Scenario* step, choose the scenario in which the result set will be used.

   b. In the *General Information* step, specify a name and description for the result set.

   c. In the *URL* step, enter the URL for the result set in your object store.

   > → Tip
   >
   > URLs must adhere to the following convention `ai://<objectStore name>/<data path>`.

   d. In the *Labels* (Optional) step, enter any labels and values that you would like to apply to the result set.

   e. In the *Review* step, confirm the details that you've provided for the result set.

5. Choose *Add* to complete result set registration.

Results

The result set that you have registered now appears in the list of result sets for the resource group.

Related Information

Manage Object Store Secrets [page 173]

6.2.4.9.2   Find a Result Set

Use the *ML Operations* app to search for a result set.

Prerequisites

You have the `artifact.register` role, or you have been assigned a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

Context

> ⚠ Caution
>
> If the files for a registered result set are deleted from your object store, or if the datapath or object store secret changes, then the registered result set can no longer be used.

Procedure

1. Choose the resource group.
2. In the *ML Operations* app, choose *Result Sets*.

   The *Result Sets* screen appears listing the result sets for the selected resource group. Result sets are listed by name, and with additional details such as description, ID, scenario, URL, execution ID, labels, created on timestamp, and changed on timestamp.
3. **Optional:** View the labels associated with any corresponding artifact by choosing ♡ *(Labels)* in the corresponding row.
4. **Optional:** Search the list by entering a result set name or part of the result set description in the
   🔍 *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.
5. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears. Enter any required filter criteria and conditions.

6.2.4.10 Other Artifacts

Any artifact that cannot be categorized as a dataset, model, or result set is registered as an "other" artifact.

An artifact is a reference to a file stored in hyperscaler object storage.

SAP AI Launchpad can be used with multiple hyperscaler object stores, such as Amazon S3, OSS, and WebHDFS.

Artifacts are unique to a resource group.

About Other Artifacts Registration

- An artifact can be manually registered using SAP AI Launchpad (see Register an Other Artifact [page 264]). You register an artifact using the *ML Operations* app. After you register an artifact, you'll see it listed with all other artifacts in the app.
- An other artifact set can be manually registered using SAP AI Core (see Create Artifacts).

Related Information

[Artifact Management](#)

6.2.4.10.1 Register an Other Artifact

Use the *ML Operations* app to manually register an "other" artifact that is stored in your object store.

Prerequisites

You have the `artifact.register` role, or you have been assigned a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

Details for the object store secret have been added using the *SAP AI Core Administration* app. See Add an Object Store Secret [page 173].

Context

> ⚠ **Caution**
>
> If the files for a registered artifact are deleted from your object store, if the datapath or object store secret changes, then the registered artifact can no longer be used.

Procedure

1. Choose the resource group. For more information, see Set Resource Group [page 164].
2. In the *ML Operations* app, choose *Other Artifacts*.

   The *Other Artifacts* screen appears listing all of the other type artifacts for the selected resource group.
3. Choose *Add* to manually register an artifact.

   The register artifact wizard appears. This wizard has five steps.
4. Enter the details for the other artifact.
   a. In the *Scenario* step, choose the scenario in which the artifact will be used.
   b. In the *General Information* step, specify a name and description for the artifact.
   c. In the *URL* step, enter the URL for the artifact in your object store.

   > → **Tip**
   >
   > URLs must adhere to the following convention `ai://<objectStore name>/<data path>`.

    d. In the *Labels* (Optional) step, enter any labels and values that you would like to apply to the artifact.

    e. In the *Review* step, confirm the details that you've provided for the artifact.

5. Choose *Add* to complete artifact registration.

Results

The artifact that you have registered now appears in the list of other artifacts for the resource group.

6.2.4.10.2 Find an Other Artifact

Use the *ML Operations* app to search for other artifacts.

Prerequisites

You have the `scenario_artifact_viewer` role, or you are assigned to a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

Context

> ⚠ Caution
>
> If the files for a registered artifact are deleted from your object store, or if the datapath or object store secret changes, then the registered artifact can no longer be used.
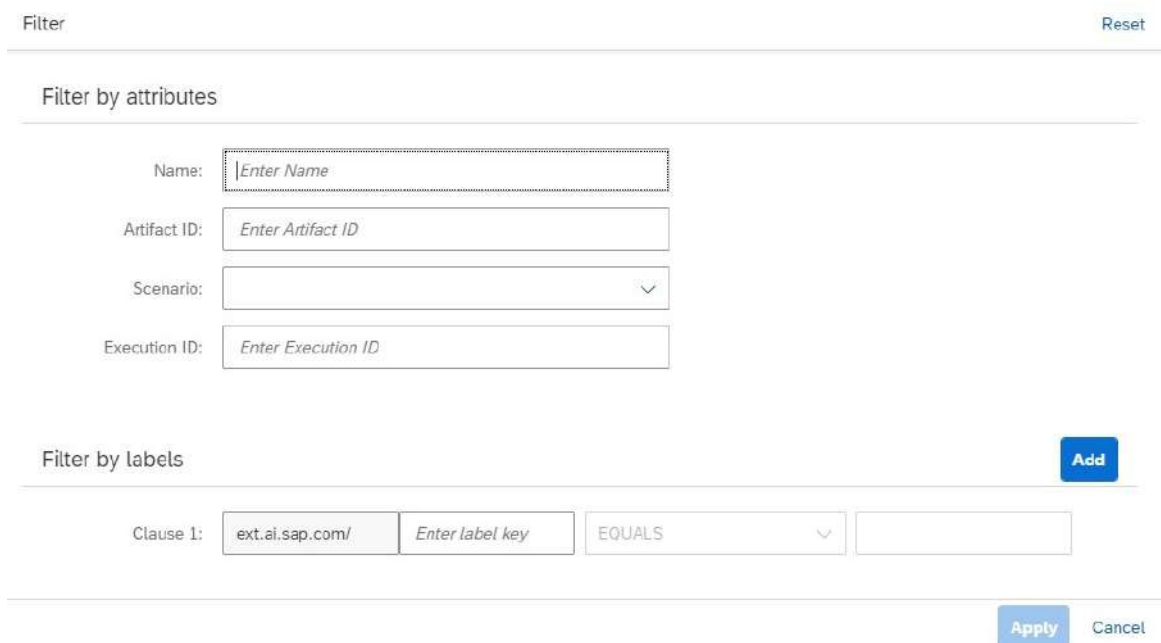
Procedure

1. Choose the resource group. For more information, see Set Resource Group [page 164].

2. In the *ML Operations* app, choose *Other Artifacts*.

    The *Other Artifacts* screen appears listing all associated artifacts. Other artifacts are listed by name, and with additional details such as description, ID, scenario, URL, execution ID, labels, created on timestamp, and changed on timestamp. Each artifact has a unique *ID*.

3. **Optional:** Search the list by entering an artifact name or part of its description in the 🔍 *(Search)* field.

> → Tip
>
> When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

4. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

a. Enter the artifact name or other attributes.
b. Choose *Add* to include the filter criteria. Adjust the conditions as needed.
c. Choose *Apply* to apply the filter and conditions to the list.
5. Check the *Labels* column for the number of classifying phrases/ names applied to the artifact. Labels are specified when registering the artifact to the selected resource group.

6.3 AI Scenario Consumer

As an AI scenario consumer, you subscribe to a service that provides AI scenarios for consumption. The services are provided as Software as a Service (SaaS) through SAP Business Technology Platform.

If you are responsible for managing the lifecycle of the AI scenario, SAP AI Launchpad provides support in the following ways:

- You can use the *Functions Explorer* app to maintain your AI scenarios.
- You can create and manage AI API connections to the AI services to which you are subscribed by using the *Workspaces* app

6.3.1 Workspaces

The *Workspaces* app is a dashboard to manage the connection between SAP AI Launchpad and subscribed AI services.

As an AI scenario consumer, you use SAP AI Launchpad to explore the AI scenarios for connected AI services.

You use the *Workspaces* app to add connections to one or more subscribed AI services, and switch between them to carry out further actions.

Accessing the App

The *Workspaces* app is available in SAP AI Launchpad from the navigation panel.
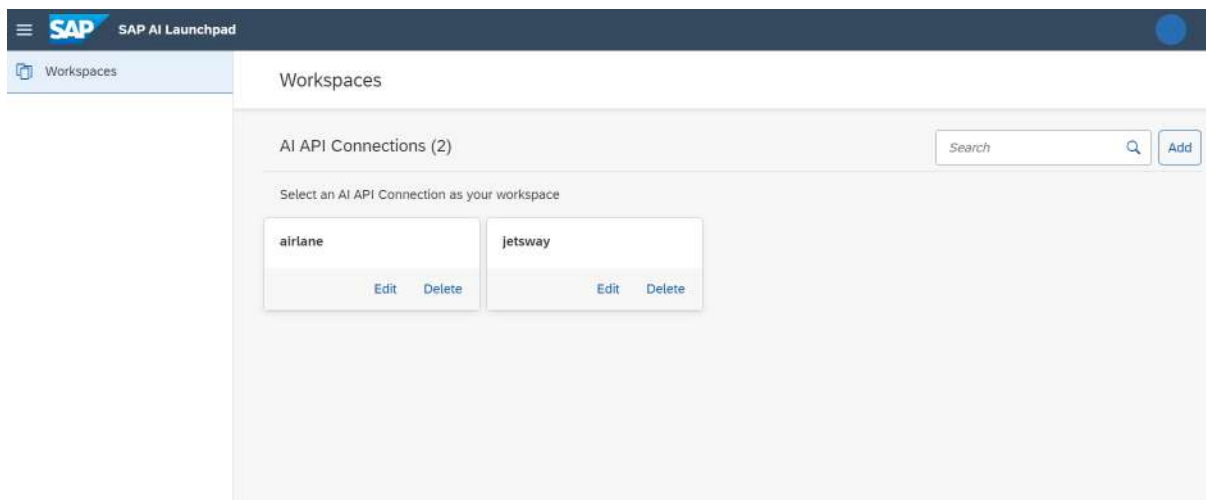
Key Tasks

Before a workspace can be used, an AI administrator must complete the following tasks:

- Add Connection [page 268]
- Manage a Connection [page 269]
- Assign Connection to Workspace [page 269]

Role Requirements

AI administrators are assigned the role `connections_editor`.

6.3.1.1 Add Connection

You add an AI API connection to link SAP AI Launchpad to your subscribed AI service.

Prerequisites

You have the `connections_editor` role or a role collection that contains it. For more information, see Roles and Authorizations [page 321].

Procedure

1. In the *Workspaces* app, choose *Add*.

   The *Create AI API Connection* dialog displays.

2. Complete the connection information for the subscribed AI service as follows:

   - *Connection Name*: Enter a name for your connection that complies with the following criteria:
     - Contains only lowercase alphanumeric characters, hyphens (-), or periods (.)
     - Starts with an alphanumeric character
     - Ends with an alphanumeric character

     > ⓘ Note
     >
     > You cannot change a connection name after it has been created.

   - Complete the remaining fields using the values provided by your AI scenario provider.

     > ⓘ Note
     >
     > Sensitive fields are masked. You can unmask to show your entry if needed.

3. Choose *Create*.

Results

The new connection appears in the *Workspaces* app. For more information, see AI API.

6.3.1.2 Manage a Connection

You can delete or edit a connection from the overview page of the *Workspaces* app.

Prerequisites

You have the `connections_editor` role, or you are assigned to a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Procedure

1. In the *Workspaces* app, find the connection.

   Each connection appears as a tile. You can search the connections by entering the name or part of the name in the Q *(Search)* field.

2. Select the connection and choose one of the following options:

   - Choose *Delete* to remove the connection from SAP AI Launchpad.

     > ⓘ Note
     >
     > Deleting a connection does not delete data associated with the service used by the connection.

   - Choose *Edit* to update the settings for the connection.

     > → Tip
     >
     > You can't change the name of a connection after it has been created. However, you can delete the connection and re-create it with the required name.
     >
     > Sensitive fields are masked and entries are not shown in clear text. To ensure these entries are accurate, copy and paste the values from your service key, or unmask the fields if needed.

6.3.1.3 Assign Connection to Workspace

Prerequisites

- You have the `connections_editor` role or you are assigned a role collection that contains it. For more information, see Roles and Authorizations [page 321].

- You have created one or more AI API connections.

Context

Use the *Workspaces* app to select a connection (subject to XSUAA roles). Once a connection is selected, the *Functions Explorer* app is available for AI operations, such as training models on the available AI use cases.

Assign Connection to Workspace

Procedure

1. In the *Workspaces* app, find the connection.

   Each connection appears as a tile. You can search the connections by entering the name or part of the name in the Q *(Search)* field.

2. Select the connection.

   The name of the selected connection appears in the title bar. You can switch between connections at any time.

   The *Functions Explorer* app is enabled.

6.3.2 Functions Explorer (Deprecated)

The *Functions Explorer* app provides tools to explore scenarios which are offered by the AI service (selected connection) to which you are subscribed. Scenarios provide AI pipelines which you use to train or deploy an AI model.

A use case is a common business challenge that can be supported with AI insights.

Within the *Functions Explorer* app, a scenario represents a use case and it is used to link the objects and processes involved in an AI pipeline .

An AI pipeline consists of the following:

- Scenarios
- Templates (run and deployment templates)
- Runs (which generate models)
- Deployments (which generate URLs for inferencing)

Role Requirements

The *Functions Explorer* app has a number of standard role collections which group commonly required roles. For more information, see Roles and Authorizations [page 321]. You can also create your own role collections and assign roles to them based on your organizational requirements.

Accessing the App

Prerequisites

- You have one of the roles described in Roles and Authorizations [page 321].
- A connection has been established as described in Assign Connection to Workspace [page 269].

Procedure

In the *Functions Explorer* app, expand the app and explore its components.
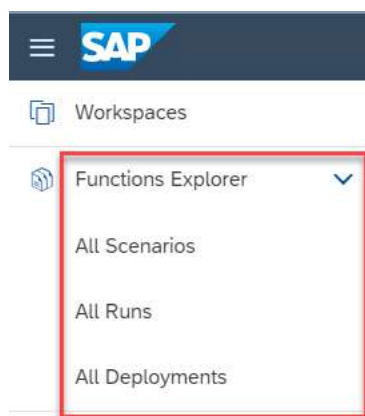
6.3.2.1   Functions Explorer Overview

As an AI scenario consumer, you subscribe to an AI service. The AI service comes with predefined AI use cases which address common business challenges.

Within the *Functions Explorer* app, a use case is called a scenario. Each scenario has a pipeline that generates or deploys a model. A pipeline consists of the objects and processes that trains or deploys a model for a specific scenario.

6.3.2.2 Scenarios

As an AI scenario consumer, you subscribe to an AI service. The AI service comes with predefined AI use cases which are called scenarios.

A scenario has a business purpose, such as determining AI insights from a recommendation system or review classification feedback.

In the *Functions Explorer* app, a scenario consists of templates that you use to implement the AI use case within your service. A scenario can have multiple versions which correspond to different template versions.

A template consists of code that defines the components required for your AI pipeline. A pipeline can be used to train a model, or to deploy a model.

Scenarios and their templates are available to all AI scenario consumers.

> ⓘ Note
>
> You use scenarios with your own input datasets and parameter values to generate models. Your datasets and generated models are unique to you and are not available to other AI scenario consumers.

6.3.2.2.1 Investigate a Scenario

In the *Functions Explorer* app, you can view a list of the scenarios available to you and explore their details.

Prerequisites

You have the `scenario_metadata_viewer` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.

   The *All Scenarios* screen appears listing all the scenarios in the resource group along with their description, created on and changed on timestamps, and number of templates and versions.

2. **Optional:** Scroll to a scenario or use Q *(Search)* to search for a scenario by name.
3. To view the details for an individual scenario, select the scenario in the list or choose ⟩ *(More)*.

   The *Scenario Details* screen appears, with the scenario name, ID, created on and changed on timestamps.
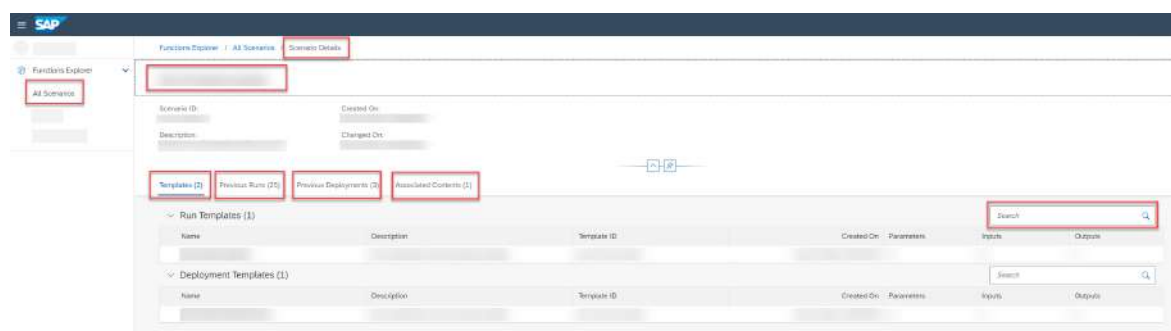
   The objects associated with the scenario can be explored in more detail using the tabs:
4. **Optional:** Investigate the scenario.

   - Choose the *Templates* tab for run and deployment template details. See Templates [page 274].
   - Choose the *Previous Runs* tab to list runs associated with this scenario. See Investigate a Run [page 300].
   - Choose the *Previous Deployments*, for deployments associated with this scenario. See Investigate a Deployment [page 311] .

- Choose the *Associated Contents*, for related input datasets and output models. See Associated Contents [page 289].

6.3.2.2.2 Templates

As an AI scenario consumer, you subscribe to an AI service. The AI service comes with predefined scenarios which contain templates.

A scenario consists of templates for the same use case. A template is an AI pipeline for a use case. A pipeline can be used to train a model, or to deploy a model.

A template consists of code that defines the components that are required for the AI pipeline.

Templates define placeholders for parameters, inputs (such as datasets), and outputs (such as models). You can use any dataset as template input, provided it has the required structure. User-defined labels can be applied to a template and are listed with the template's details.

A configuration helps you attach values to the placeholders for the template. You can create multiple configurations for the same template.

There are two types of template; run template and deployment template.

- A template that is used to train a model is called a run template. An instance of a run template is a run. Runs generate output, such as models.
- A template that is used to deploy a model is called a deployment template. An instance of deployment template is a deployment. Deployments generate URLs that can be used for inference.

To use a template, you must create its instance (either a run or a deployment) and associate with a configuration (data).

> ⓘ Note
>
> Templates are re-usable. The same template can be used for either training or deploying a model.

> ⚙ Example
>
> Consider a scenario, such as a product review classification (involving positive or negative reviews). The run template `Product_Review_Training` has placeholders, as follows:
>
> Input parameters:
>
> - `Num_of_epochs`
> - `Max_Depth_Decision_Tree`
>
> Input (datasets):
>
> - `Stopwords_Dataset`
> - `Past_Reviews_Dataset`

The following types of template are available in a scenario for AI scenario consumers:

Run Templates [page 275]
    A template that is used to train an AI model is called a run template .

Deployment Templates [page 281]

A deployment template is used to deploy an AI model.

6.3.2.2.2.1 Run Templates

A template that is used to train an AI model is called a run template .

A run template defines the following:

- Parameters
  Hyperparameter values (variables that affect the decision-making capability of a model)
- Inputs
  The dataset on which a model is trained, see Datasets [page 289]
- Outputs
  The model that is the result of the training process, see Models [page 290]

Values for parameters and other inputs are provided by a configuration. To start the training process with these values, a run is created with the configuration.

6.3.2.2.2.1.1 Investigate a Run Template

Use the *Functions Explorer* app to view a list of run templates, and explore a template in detail.

Prerequisites

You have the `scenario_executable_viewer` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.
2. Find the scenario and display its details. For more information, see Investigate a Scenario [page 273].

   The *Scenario Details* screen contains a *Templates* tab, which displays the run and deployment templates associated with the scenario, as well as summary details such as template ID, created on date, and number of parameters, inputs, and outputs.
3. To view the details for a run template, double-click on a template in the list.

   The *Template Details* screen appears, with the template ID, description, scenario version, created on timestamp, parameters, input dataset, output model, and labels.
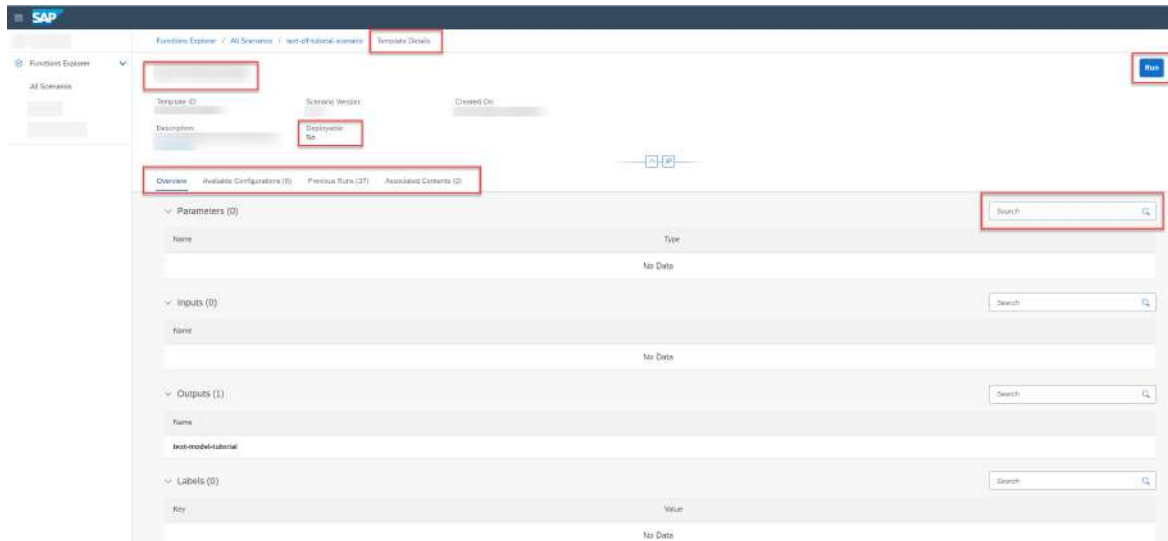
> → Tip
>
> Check the value of the *Deployable* field to confirm whether the template is a run or deployment template. Run templates are **not** deployable.

4. **Optional:** Select the *Available Configurations* tab to show the configurations associated with this template. See Find a Configuration [page 276].

5. **Optional:** Select the *Previous Runs* tab to show the runs associated with this template. See Find a Run [page 278].

6. **Optional:** Select the *Associated Content* tab for datasets and models associated with this template. See Associated Contents [page 289].

7. **Optional:** To create a new run, choose *Run*. See Create a Run [page 279].

Results

You can use the run template to find the objects associated with the scenario and template.

6.3.2.2.2.1.2 Find a Configuration

You can view all the configurations associated with a run template, and investigate a configuration in detail.

Prerequisites

You have the `scenario_configuration_viewer` or `scenario_configuration_editor` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A configuration is required to create a run. The configuration provides values for parameters and other inputs that the run template has defined. Multiple configurations can be associated with a run template.

Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.
2. Find the run template and display its details. For more information, see Investigate a Run Template [page 275].
3. On the *Template Details* screen, select the *Available Configurations* tab.
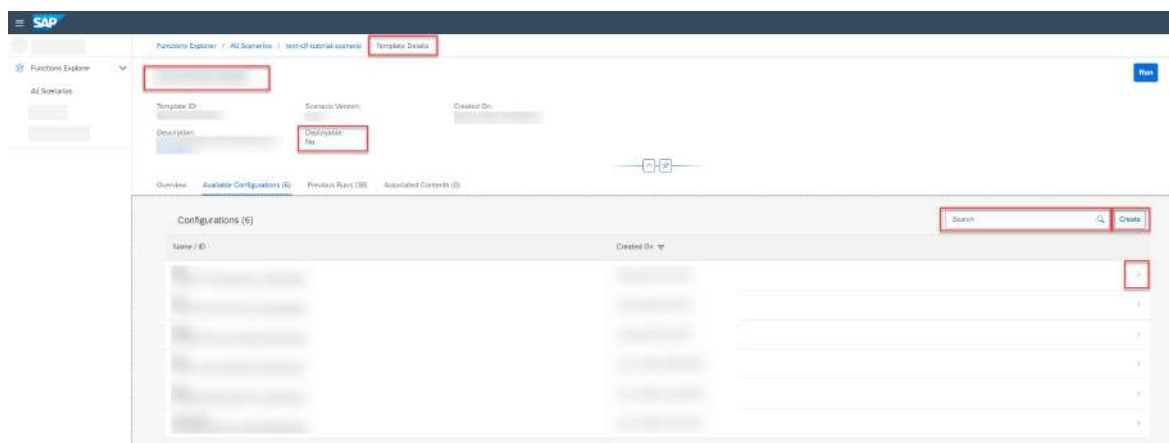
   All configurations associated with the run template are listed, as well as details such as name, ID, and date of creation.

4. **Optional:** Search the list by entering a configuration name or partial name in the Q *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

5. **Optional:** To create a new configuration, choose *Create*. See Create a Configuration [page 280].
6. To view the details for a configuration, select a configuration in the list or choose ❯ *(More)*.
   The *Configuration Details* screen appears with the configuration name, configuration ID, scenario, template, created on timestamp, parameters, and inputs. To copy the configuration ID, click 🗐 *(Copy)*.

6.3.2.2.2.1.3 Find a Run

You can find a run and explore its details.

Prerequisites

You have the roles `scenario_job_viewer` and `scenario_metric_viewer`, or a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A run is a training process which generates a model. A run is created using a configuration and a run template.

A run's details consists of two tabs: *Overview* and *Metric Resource*.

- The *Overview* tab displays the scenario and template, as well as the parameters, input dataset and output model associated with the run.
- The *Metric Resource* tab displays the performance metrics and tags for the generated model.

> → Tip
>
> In the *Functions Explorer* app, there are multiple ways to investigate a run's details. You can find a run via it's scenario and template, or search for it via *All Runs*.

Procedure

1. In the *Functions Explorer* app, find and display a run template. See Investigate a Run Template [page 275].

   The *Template Details* screen appears with overview, configuration, and deployment details for the selected run template.

2. On the *Template Details* screen, select the *Previous Runs* tab.

   The list shows all previous runs for the run template, as well as key details such as run ID, configuration name and ID, current and target status, created on and changed on timestamps.

3. **Optional:** To compare metrics for up to five runs, see Compare Run Metrics [page 305].

4. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.
   a. Enter the run ID or select a status.
   b. Choose *Apply* to apply the filter to the list.

5. To view the details for a run, select a run in the list or choose ⟩ *(More)*.

   The *Overview* tab displays details associated with the run, such as run ID, names of the associated scenario and template, parameters, input dataset, and output model.

> **→ Tip**
>
> To find the model generated by the run, choose ▣ *(Copy)* to copy the model ID. Then see Investigate a Model [page 291].

The *Metric Resource* tab displays standard evaluation metrics and associated labels, tags, and custom info, which are stored and logged by the run template. You can compare metrics between different runs if required.

Related Information

Compare Run Metrics [page 305]
Investigate a Run [page 300]
Runs [page 298]

6.3.2.2.2.1.4 Create a Run

A run is a training process that uses a run template and a configuration to generate a model.

Prerequisites

You have the `scenario_job_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Context

Runs are used to train a model.

Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.
2. Find the run template and display its details. For more information, see Investigate a Run Template [page 275].
3. Choose *Run* to create a new run.

   The *Create New Run* wizard appears. This wizard has four steps.

4. Select the required data for the new run. Note, the scenario and run template are prefilled from your previous selections.

   1. In the *Select Configuration* step, select the required configuration. The details for the selected configuration are displayed in the right pane. Confirm the selection and choose *Review*.

      > → Tip
      >
      > If there is no configuration which matches your data requirements, you can choose *create a configuration*. The run you have started will be lost, and you will be redirected to create a configuration. When you have saved the new configuration, you can re-create the run with the new configuration. See Create a Configuration [page 280].

   2. In the *Review* step, review the data that you've selected for the new run. Choose *Create* to create the run.

   The new run is created.

Related Information

Find a Run [page 278]
Investigate a Run [page 300]
Runs [page 298]

6.3.2.2.2.1.4.1 Create a Configuration

A configuration contains parameters and dataset references, and is combined with a run template to create a run.

Prerequisites

You have the `scenario_configuration_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Procedure

1. Find the run template. For more information, see Investigate a Run Template [page 275]
2. Select the *Available Configurations* tab.

   This lists all configurations by name and ID, and with creation date details.

3. Choose *Create* to create a new configuration.

   The *Create Configuration* wizard appears. This wizard has four steps.

4. Enter the required data for the new configuration.

   1. In the *Enter Name* step, enter a configuration name and choose *Next*.
   2. In the *Input Parameters* step, enter the input parameter values (alphanumeric) for the selected job executable, then choose *Next*.

      > ⚠ **Restriction**
      >
      > A configuration can contain up to 1000 input parameters. An input parameter value can't exceed 5000 characters.

   3. In the *Input Contents* step, find the name of the dataset in the list of *Available Contents* list (right pane). To search for a dataset, enter a value or partial value in the 🔍 *(Search)* field.
      For the selected dataset, choose the dropdown for the *Assignment* field and select the input.
   4. Choose *Review* to display a summary of the configuration details. Choose *Create* to create the configuration.

   The newly created configuration and its details are displayed.

Related Information

6.3.2.2.2 Deployment Templates

A deployment template is used to deploy an AI model.

A deployment template defines placeholders for key data such as:

- Model (in most cases)
- Parameters
- Hyperparameter values for the model (variables that affect the decision-making capability of the model)
- Efficiency parameters

Values for parameters and other inputs are provided by a configuration. To start the deployment process with these values, a deployment is created with the configuration.

Efficiency Features in the SAP AI Core Runtime

In addition, the SAP AI Core runtime that improve model server efficiency and help manage resource consumption during the deployment.

Autoscaling

SAP AI Core includes parameters to reduce the number of nodes used based on current consumption, or impose usage limits during periods of high consumption. These parameters allow your workload the flexibility to scale based on demand, and for consumption to be capped, limiting your consumption and therefore costs. For more information, see Serving Templates.

Scaling to 0

Where non-uniform loads are expected, scaling to 0 allows nodes to enter a sleeping state when demand allows, limiting your consumption, and therefore costs. Nodes wake when demand increases, which has an increased response time. The Global Node pool reduces, this cold start time. For more information, see Serving Templates.

Global Node Pool

When the inference server scales up from a sleeping state, there is some additional response time. To reduce this, SAP AI Core has a Global Node Pool, which keeps commonly used nodes reserved to allow for shorter response times. You do not need to do anything to make use of the Global Node Pool, it is already in place. To reduce response times further, avoid a cold start altogether by setting your autoscaling parameter to 1. For more information, see Serving Templates.

Scaling to 1

Cold starts can be avoided completely by scaling to 1. This keeps a single node warm, even when it is not needed, reducing response time. However, it does not offer the consumption and cost savings associated with scaling to 0. For more information, see Serving Templates.

Duration

The default duration is indefinite, however the `ttl` parameter limits the duration of a deployment to minutes, hours, or days. This parameter allows you to plan the deletion if your model servers and model deployment

URL, allowing for an expected period of use and avoiding unnecessary consumption and costs afterwards. For more information, see Deploy Models and About the AI API.

6.3.2.2.2.2.1 Investigate a Deployment Template

Use the *Functions Explorer* app to view a list of deployment templates, and explore a template in detail.

Prerequisites

You have the `mlfunctions_viewer` or `mlfunctions_editor` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

## Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.
2. Find the scenario and display its details. For more information, see Investigate a Scenario [page 273].

   The *Scenario Details* screen contains a *Templates* tab, which displays the run and deployment templates associated with the scenario, as well as summary details such as template ID, created on date, and number of parameters, inputs, and outputs.

3. To view the details for a deployment template, double-click on a deployment template in the list.

   The *Template Details* screen appears, with the template ID, description, scenario version, created on timestamp, parameters, input dataset, output model, and labels.
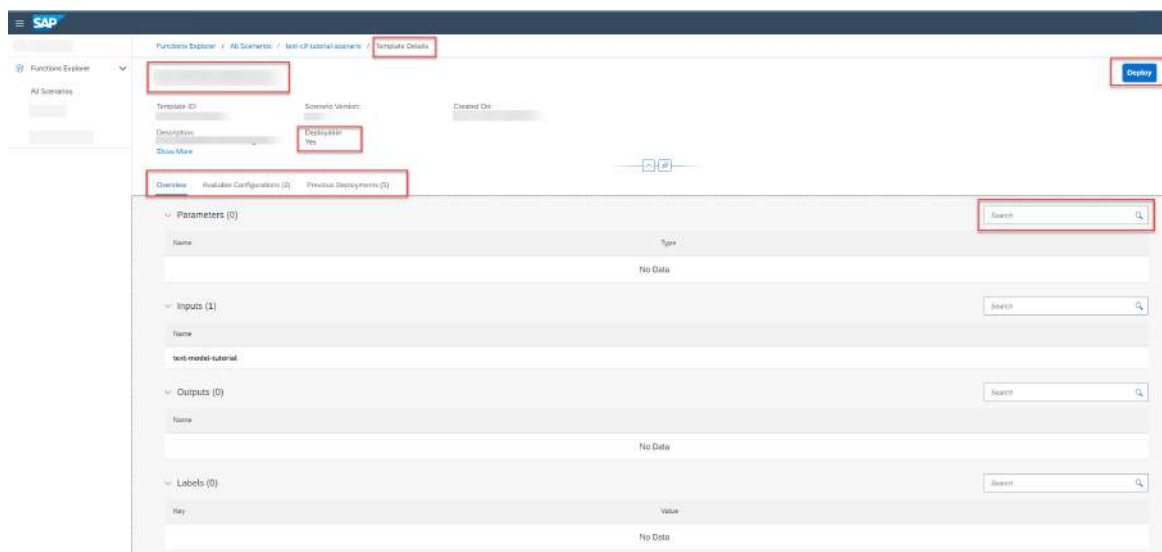
> → Tip
>
> Check the value of the *Deployable* field to confirm whether the template is a run or deployment template. Deployment templates are deployable (`Yes`).

4. **Optional:** Select the *Available Configurations* tab to show the configurations associated with this template. See Find a Configuration [page 284].
5. **Optional:** Select the *Previous Deployments* tab to show the deployments associated with this template. See Find a Deployment [page 285].
6. **Optional:** To create a new deployment, choose *Deploy*. See Create a Deployment [page 287].

Results

The deployment template details page appears, with the name and timestamp details for the template, the dataset, and associated configurations and deployments.

6.3.2.2.2.2.2 Find a Configuration

You can view all the configurations associated with a deployment template, and investigate a configuration in detail.

Prerequisites

You have the `mlfunctions_viewer` or `mlfunctions_editor` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A configuration is required to create a deployment. Multiple configurations can be associated with a deployment template.

Procedure

1. In the *Functions Explorer* app, find the deployment template and display its details. For more information, see Investigate a Deployment Template [page 283].

2. On the *Template Details* screen, select the *Available Configurations* tab.

   All configurations associated with the deployment template are listed, as well as details such as configuration name and ID, and date of creation.
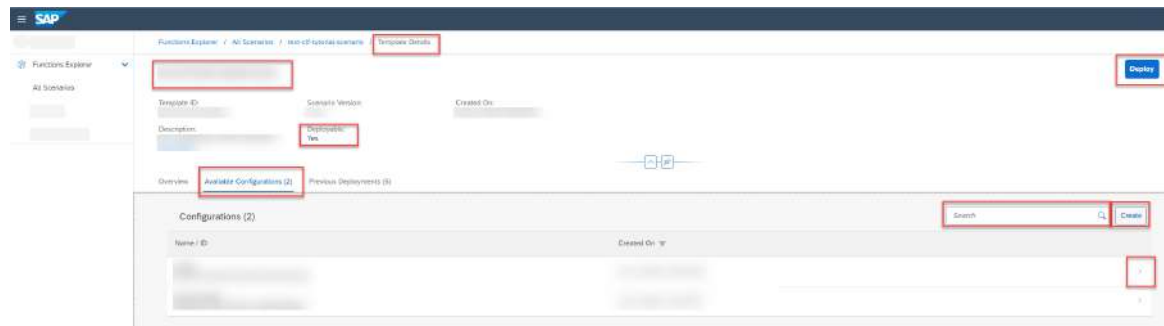
3. **Optional:** Search the list by entering a configuration name or ID in the 🔍 *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

4. **Optional:** To create a new configuration, choose *Create*. See Create a Configuration [page 288].

5. To show the details for a configuration, select a configuration in the list or choose ❯ *(More)*.
   The *Configuration Details* screen appears with the configuration name, configuration ID, scenario, template, created on timestamp, parameters, and inputs. To copy the configuration ID, click 🗐 *(Copy)*.

6.3.2.2.2.3 Find a Deployment

You can find a deployment and display its details.

Prerequisites

You have the `mlfunctions_viewer` or `mlfunctions_editor` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A deployment is a process which deploys a model. A deployment is created using a configuration and a deployment template. Multiple deployments can be created for a deployment template.

> → Tip
>
> In the *Functions Explorer* app, there are multiple ways to investigate a deployment's details. You can find a deployment via it's scenario and deployment template, or search for it via *All Deployments*.

Procedure

1. In the *Functions Explorer* app, find and display a deployment template. See Investigate a Deployment Template [page 283].

   The *Template Details* screen appears with overview, configuration, and deployment details for the selected deployment template.

2. Select the *Previous Deployments* tab.

   The list displays all previous deployments for the template, with deployment ID, configuration name and ID, current and target status, and creation and change on dates.
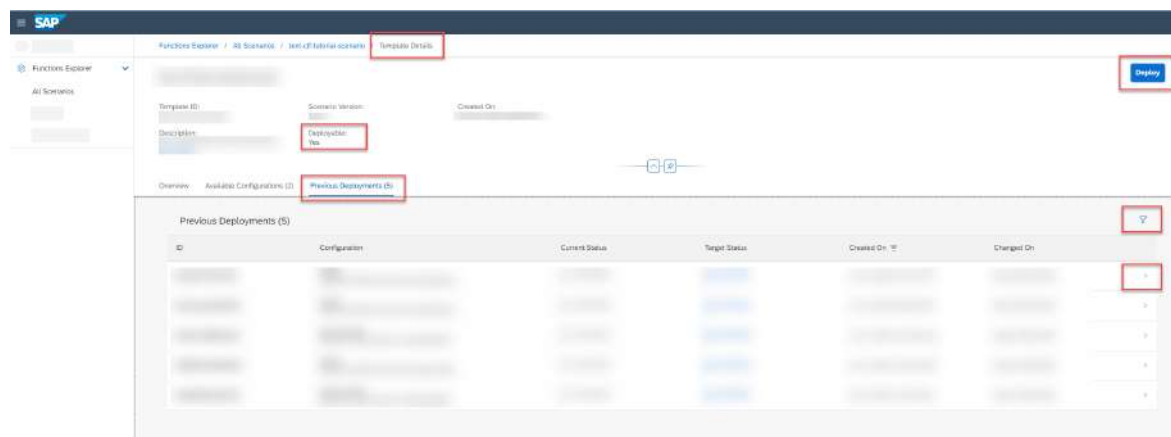
3. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.
   a. Enter the deployment ID or status.
   b. Choose *Apply* to apply the filter to the list.

4. **Optional:** To create a new deployment using the deployment template, choose *Deploy*. See Create a Deployment [page 287].

5. To investigate details for a deployment, select the deployment in the list or choose ＞ *(More)*.

   The *Deployment Details* screen appears with details such as the current and target status, scenario, deployment template, parameters, and dataset.

6.3.2.2.2.2.4 Create a Deployment

A deployment uses a model and data to make a prediction.

Prerequisites

You have the `mlfunctions_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.
2. Find the deployment template and display its details. See Investigate a Deployment Template [page 283].
3. Choose *Deploy* to create a new deployment.

   The *Create New Deployment* wizard appears. This wizard has four steps.
4. Select the required data for the new deployment. Note, the scenario and deployment template are prefilled from your previous selections.
   1. In the *Select Configuration* step, select the required configuration. The details for the selected configuration are displayed in the right pane. Confirm the selection and choose *Review*.

      > → Tip
      >
      > If there is no configuration which matches your data requirements, you can choose *create a configuration*. You will be redirected to create a configuration and your deployment will be lost. When you have saved the new configuration, you can re-create the deployment with the new configuration. See Create a Configuration [page 288].

   2. In the *Review* step, confirm the data that you've selected for the deployment. Choose *Create* to create the deployment.

   The new deployment is created and its details displayed.

6.3.2.2.2.2.4.1 Create a Configuration

A configuration contains parameters and dataset references, and is combined with a deployment template to create a deployment.

Prerequisites

You have the `mlfunctions_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Procedure

1. Find the deployment template. See Investigate a Deployment Template [page 283].
2. Select the *Available Configurations* tab.

   The list shows all configurations by name and ID, and with creation date details.
3. Choose *Create* to create a new configuration.

   The *Create Configuration* wizard appears. This wizard has four steps.
4. Enter the required data for the new configuration.

   1. In the *Enter Name* step, enter a configuration name and choose *Next*.
   2. In the *Input Parameters* step, enter the input parameter values (alphanumeric) for the selected template, then choose *Next*.

      > ⚠ Restriction
      >
      > A configuration can contain up to 1000 input parameters. An input parameter value can't exceed 5000 characters.

   3. In the *Input Contents* step, find the name of the model in the list of *Available Contents* list (right pane).
      To search for a model, enter a value or partial value in the 🔍 *(Search)* field.
      For the selected model, choose the dropdown for the *Assignment* field and select the input.
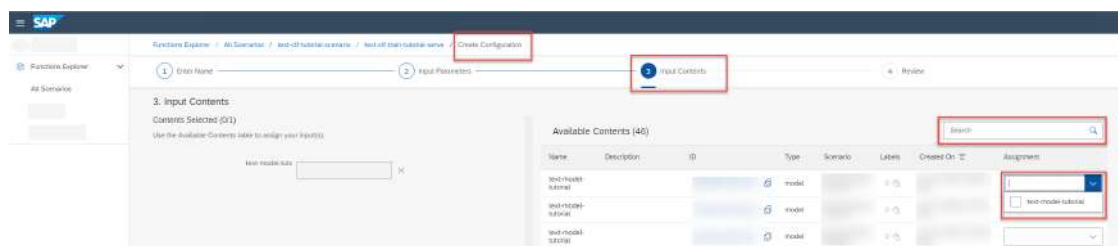
   4. Choose *Review* to display a summary of the configuration details. Choose *Create* to create the configuration.

   The newly created configuration and its details are displayed.

Related Information

6.3.2.2.3   Associated Contents

6.3.2.2.3.1 Datasets

A dataset provides the data used to train a model. Datasets are files which are stored in a hyperscaler storage bucket (such as AWS S3).

Find a Dataset

Prerequisites

You have the `scenario_artifact_viewer` role, or you have been assigned a role collection that contains this role.

For more information, see .

Procedure

1. In the *Functions Explorer* app, choose *All Scenarios*.
2. Find the scenario and display its details. For more information, see .
3. Select the *Associated Contents* tab to display datasets and models associated with the scenario.
4. Check the list of available datasets.
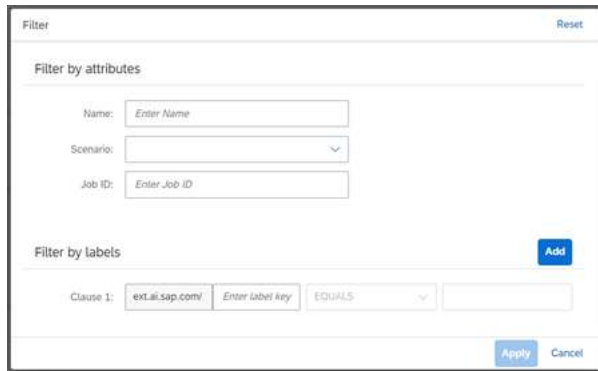
   The columns show the following:

   - The names of the datasets
   - A unique identifier for each dataset
   - The scenario associated with the dataset
   - Custom tags (labels) that were associated with the dataset when it was created. You can view the labels associated with an dataset by choosing ⬙ *(Labels)*in the corresponding row.
   - The timestamp when the dataset was created

5. **Optional:** Search the list by entering a dataset name or part of the dataset description in the ⚲ *(Search)* field.

> **→ Tip**
>
> When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

6. **Optional:** Filter for a dataset in the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

a. Enter either the dataset name or other attributes.
b. Choose *Add* to include the filter criteria. Adjust the conditions as needed.
c. Choose *Apply* to apply the filter and conditions to the list.

6.3.2.2.3.2 Models

A model is an algorithm that learns patterns from a given set of training data (dataset) to accomplish a certain task. Models are generated by a training process. A model consists of files stored in a hyperscaler storage bucket (such as AWS S3).

When you create a run and train a model, a model is generated in the form of files. These files are written to the hyperscaler storage bucket connected to your subscribed AI service (that is, your selected connection). Each model has a unique model ID.

You use the *Functions Explorer* app to find and display a model, at either scenario or run template level.

As a business analyst using SAP AI Launchpad, you can generate models with a high degree of accuracy. You assess a model's quality and accuracy by evaluating the data on the *Metrics* tab. Metrics data is generated by the tun template during the training process. The datasets and hyperparameters selected affect the metrics (quality of the model). For more information, see View Metrics for a Model [page 292].

When a model reaches the required quality, you can share the hyperparameters with the ML Ops users of your organization, who can then retrain the model on production datasets and deploy the model.

6.3.2.2.3.2.1 Investigate a Model

A model consists of overview and metric data.

Prerequisites

You have the `scenario_artifact_viewer` and `scenario_metric_viewer` roles, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A model consists of both overview and metric data. Overview data summarizes the labels, parameters, and inputs associated with the source run. Metric data is affected by the dataset and parameter values used by the run, and can be used to determine a model's quality.

You can find and display a model at either scenario or run template level. If you have copied a model's ID, you can also search for the model.

You can create a chart for an individual model, or create a chart to compare models.

Procedure

1. Find the run template. For more information, see Investigate a Run Template [page 275].
2. On the *Run Details* screen, select the *Associated Contents* tab.

   Objects associated with a run template are filtered by type, such as models or datasets.
3. Select **Models** in the *Show* field to display all the models generated by the run template.
4. **Optional:** Search the list by entering a model ID or partial value in the 🔍 *(Search)* field.

   > → Tip
   >
   > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

5. **Optional:** To create a chart for a model, select a model in the list and choose *View Metrics*. See Create Chart to Compare Models [page 293].
6. To view the details for a model, select a model in the list or choose 〉 *(More)*.
   The *Model Details* screen appears with key details, such as model ID and description, scenario and template, created and changed on timestamps, and source run.
7. Check the *Overview* tab for labels, parameters, and inputs associated with the source run.
8. Check the *Metrics* tab to evaluate the quality of the model. See View Metrics for a Model [page 292].

Related Information

6.3.2.2.3.2.2 View Metrics for a Model

A metric provides a measure of quality or confidence for a model.

Prerequisites

You have the `scenario_artifact_viewer` or `scenario_metric_viewer` role, or you are assigned to a role collection that contains one of these roles. For more information, see Roles and Authorizations [page 321].

Context

During a run, metrics (standard model evaluation metrics and associated labels, tags, and custom info) are stored and logged by the run template. You review metrics to evaluate the quality of a model generated by the run. Metric quality is affected by the input dataset and parameter values used.

> ⓘ Note
>
> Metric data is saved to both models and their source runs. This means you can also view the same metric data for the source run. A metric resource is a collection of all tracked metrics for a run, including labels and tags. See View the Metric Resource for a Run [page 302]. For information about the types of data stored, see Storing Metric Data.

Procedure

1. In the *Functions Explorer* app, find the model and display its details See *Investigate a Model* [page 291].
2. Select the *Metrics* tab to check the metrics and values which were captured when training the model.

   > ⓘ Note
   >
   > The *Metrics* tab is only visible if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

   Metric details are displayed, as follows:
   - *Name*: Quality criteria. For example, accuracy or mean absolute error (MSE).

- *Value*: Indicates a level of quality, and is dependent on the *Name* (criteria).
- *Timestamp* and *Step*: Used to uniquely identify or differentiate the results. For metrics that are logged multiple times, the timestamp and step can be used in conjunction to check how a metric has progressed during the run (training process). For example, a model trains iteratively on the same dataset in a single training process. Also known as an epoch.
- *Labels*: Custom information associated with the metric for that run template.

Related Information

Compare Model Metrics [page 295]

6.3.2.2.3.2.3 Compare Models

You can compare models to determine which configuration parameters result in optimum results.

You can compare up to five models, either by comparing metric data or by creating charts.

- Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the run (training process) and by the input configuration parameters.
  When you compare metrics for runs, multiple run metrics are compared against input configuration parameters. You compare metric data to determine whether further adjustments are required for configurations, or to identify a configuration which is producing optimum results.
- Charts provide a visual representation of a run and resulting model quality. You can choose the chart settings and chart type to graphically compare models. Chart types include line, bar, column, heat map, and scattered charts. The chart types available for use depend on your source data and chart settings. You can preview a chart before you add it to your chart view.

> ⓘ Note
>
> You can only make comparisons if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

6.3.2.2.3.2.3.1 Create Chart to Compare Models

You can create charts for models to visually compare quality criteria and values.

Context

You create a chart to visualize metrics for:

- A single model, or

- Multiple models, comparing up to five models against timestamp, step, or configuration parameters

You can find and compare models at either scenario or run template level. You can create multiple charts and view them in the *Chart View*.

> ⓘ Note
>
> Each chart can compare data for up to five models.

Procedure

1. In the *Functions Explorer* app, find and display a scenario. See Investigate a Scenario [page 273].
2. List the models for comparison:

   - To list models for a scenario, select the *Associated Contents* tab.
   - To list models for a run template, find and display the run template. See Investigate a Run Template [page 275]. Then select the *Associated Contents* tab.
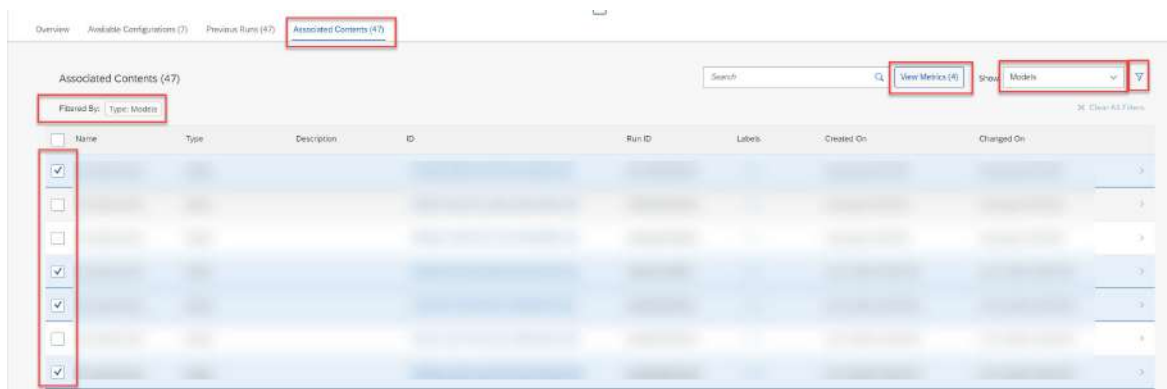
3. **Optional:** Select `Models` in the *Show* field to show the models generated by the run template.
4. **Optional:** If you have a large list of models, you can further filter the list by choosing ▽ *(Filter)*.
5. Select the models for comparison and choose *View Metrics*.

   The *Metrics Overview* appears for the selected models. The model IDs and descriptions are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.
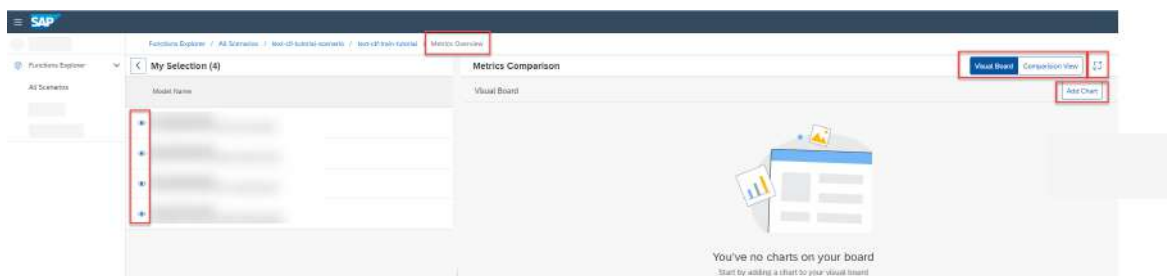
6. Choose *Add Chart* to create a chart comparing the selected models.

   The *Add Chart* dialog appears.
7. Enter the chart settings:

- Enter a name and description for the chart.
- In *Chart Settings*, choose **Models** as the metrics source.
- In *Comparison Type*, choose your preferred comparison. You can compare metrics to parameters, to the source, or to steps or time. Based on your selection, you'll be prompted to select the metrics and values for comparison.

8. Choose *Preview* to continue to the preview settings.

> ⚠ Caution
>
> If *Preview* is not enabled, review your settings and selections. Some settings are mandatory, and you can't proceed until specified. Some settings and data combinations don't correspond to a valid chart type.

9. In the *Chart Selection* pane, select the chart type (such as column or bar chart). Note, the chart types available depend on the chart settings you defined.

10. Choose *Data Selection* from the dropdown menu to confirm the models selected for the chart.

    You can show or hide models from the data selection, and see the impact on the preview chart.

11. Choose *OK* to create the chart. The chart appears in your chart view.

12. **Optional:** Check the chart. Note, if you have multiple charts, you may need to scroll.

    a. To display a chart in full-screen mode, choose ⤢ *(Open Full Screen)*.

    b. To edit a chart, choose ✎ *(Edit)*.

    c. To delete a chart from your chart view choose ✎ *(Delete)*.

6.3.2.2.3.2.3.2 Compare Model Metrics

You can compare metrics for models to determine which configuration parameters result in optimum results.

Context

> ⓘ Note
>
> You can compare metrics for up to five models.

Procedure

1. In the *Functions Explorer* app, find the run template. For more information, see Investigate a Run Template [page 275].

2. On the *Run Details* screen, select the *Associated Contents* tab.

    Objects associated with a run template are filtered by type, such as models or datasets.

3. Select **Models** in the *Show* field to display all the models generated by the run template.

4. **Optional:** Filter the list by entering the model name, model ID, or run ID in the ∇ *(Filter)* .

5. Select the models for comparison and choose *View Metrics*.
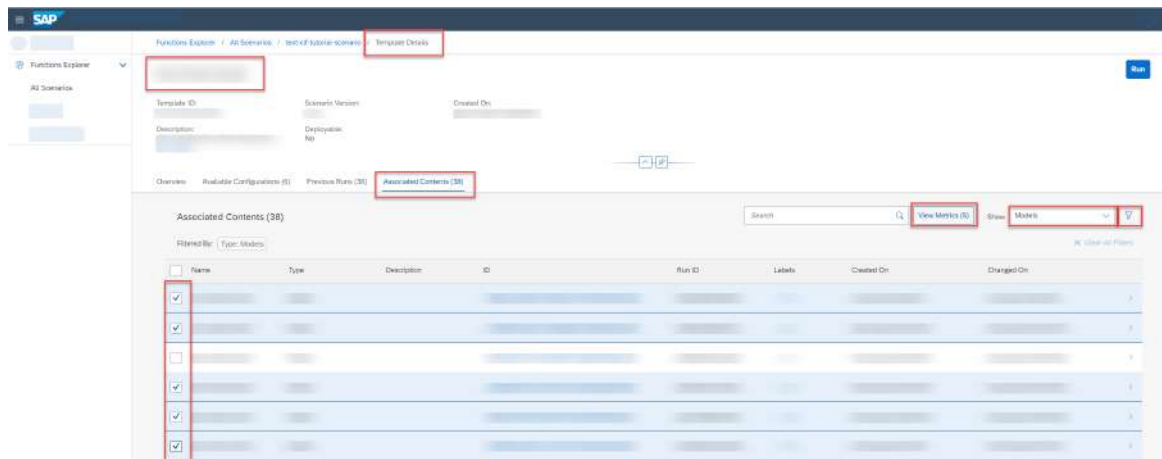
The *Metrics Overview* screen appears for the selected models. The model IDs and descriptions you selected are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.

6. Choose *Comparison View* to compare metric data.

7. Investigate the metric data for the selected models.

- To show the comparison results in full screen mode, choose ⛶ *(Fullscreen)*.
- To show or hide the models selected for comparison, choose ⊙ *(Show)* or ⊘ *(Hide)*.
- To show or hide different comparison data, choose *Configurations*, *Executions*, or *Models*.
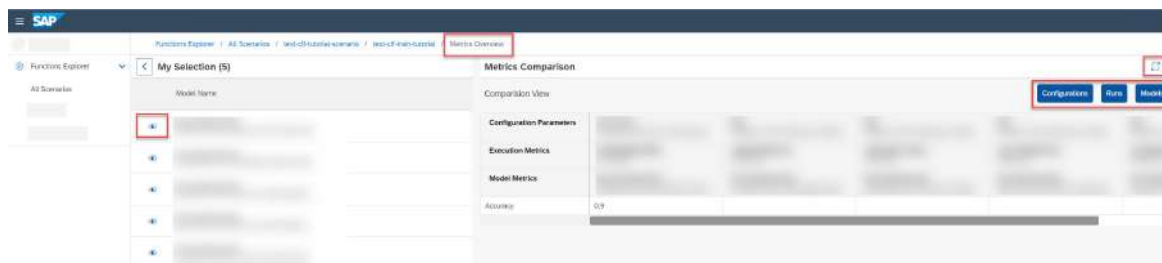
Related Information

[Create Chart to Compare Models [page 293]](#)

6.3.2.2.3.2.4 Train a Model

In the *Functions Explorer* app, you create a run to train a model.

Prerequisites

You have the `scenario_job_editor` role, or you are assigned a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

Context

The steps to train a model and create a run are the same. A run is the training process that generates a model.

Training a model is an iterative process. You create successive runs with different values for the datasets and parameters in the run template. A configuration combines the dataset values for input, and values for parameters. The quality of the resulting model depends on the specified datasets and parameters in the configuration. You can expect to experiment iteratively with these values until your model reaches the required benchmark.

To determine how configuration parameters are affecting your models, see Compare Model Metrics [page 295].

Procedure

1. In the *Functions Explorer* app, find the run template. For more information, see Investigate a Run Template [page 275].

   The *Template Details* screen appears. From here, you can display the configurations, previous runs, and associated contents (datasets and models) for the run template.

2. Choose *Run* to create a new run and generate a new model.
   A wizard appears to assist with the settings for the run. The scenario and run template values are pre-filled based on your previous selections.

3. Select a configuration.

   - Search for a configuration in the list by entering a name or part of a name in the 🔍 *(Search)* field.

     > → Tip
     >
     > When your runtime is SAP AI Core, this search is not case-sensitive. For other runtimes, search may be case-sensitive.

   - Select a configuration and confirm the settings using the preview (right) pane.

   - If the existing configurations don't have your required parameters, choose *create a configuration*. See Create a Configuration [page 280].

4. Choose *Review* to review the details of the run.
5. Check the settings and choose *Create* to create a new run.
   A run is created and started.
6. Choose *Refresh* to update the status of the job.

   When the run is complete, a model is generated as output. You can compare the model with other models to determine its quality. See Compare Model Metrics [page 295].

6.3.2.3 Runs

A run is a training process that generates a model or models. A run is an instance of a run template (an AI pipeline), created using a configuration.

About Runs

A run is used to train a model or for other AI workflows.
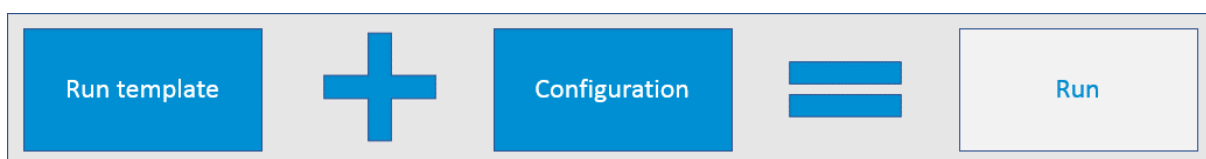
- A run template defines the expected parameters, input dataset, and output model that is generated from the training process.
- Values for parameters and input dataset are provided by a configuration.
- A run is created to start the training process using the provided values.
- A run generates a model.

Runs are associated with a large volume of metadata, as well as training/ model performance, and other training data. Metadata is stored as metrics (and associated labels), tags, and custom info. Multiple metrics, tags and custom info can be stored with a single run. Once metric, tag and custom info are saved to a run, they can be queried with a run ID.

You can view runs and individual run details in the *Functions Explorer* app, as follows:

- To see runs for all scenarios, navigate to the *All Runs* screen.
- To see runs for a specific scenario, navigate to the *All Scenarios* screen. Choose a scenario and select the *All Runs* tab.

For more information, see Investigate a Run [page 300].
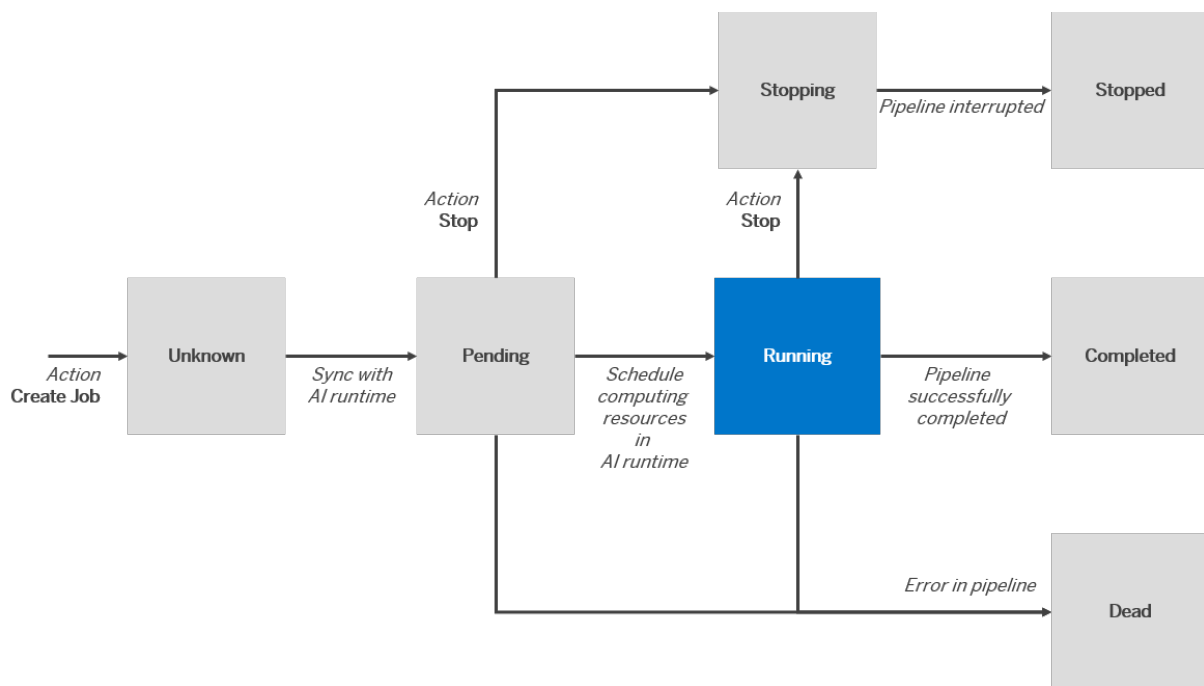
Runs and Models

You can expect to experiment iteratively with the datasets and configuration parameters until your model reaches the expected benchmark.

Run States

Runs can have any of the following states:

Pending
Running
Stopping
Stopped
Completed
Dead
Unknown

The following figure shows how run states can change following the initial state of *Pending*:

Run State Flow

The AI runtime refers to the infrastructure of your subscribed AI service. When a run state changes from *Running* to the next state, the computing resources that were used by the AI runtime are released.

Stop/Delete Behavior by State for Runs

State	Stop Run	Delete Run
Unknown	Not enabled	Enabled

State	Stop Run	Delete Run
Pending	Enabled	Not enabled
Running	Enabled	Not enabled
Completed	Not enabled	Enabled
Stopping	Not enabled	Not enabled
Stopped	Not enabled	Enabled
Dead	Not enabled	Enabled

6.3.2.3.1 Investigate a Run

You can explore run details for detailed insights about the training process for a model.

Prerequisites

You have the scenario_job_viewer role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Context

A run's details consists of two tabs: *Overview* and *Metric Resource*.

- The *Overview* tab displays the scenario and template, as well as the parameters, input dataset and output model associated with the run.
- The *Metric Resource* tab displays the performance metrics for the generated model.

Procedure

1. In the *Functions Explorer* app, choose *All Runs*.

   The *All Runs* screen appears listing all of the runs by ID, and with additional details such as configuration name and ID, current and target status, created on and changed on timestamps.

2. **Optional:** To compare metrics for up to five runs, see Compare Run Metrics [page 305].

3. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

   a. Enter the run ID or select a status.
   b. Choose *Apply* to apply the filter to the list.

4. To view the details for a run, select a run in the list or choose ❯ *(More)*.

The *Overview* tab displays details associated with the run, such as run ID, names of the associated scenario and template, parameters, input dataset, and output model.

> → Tip
>
> To copy the ID of the input dataset or output model, choose 🗐 *(Copy)*.
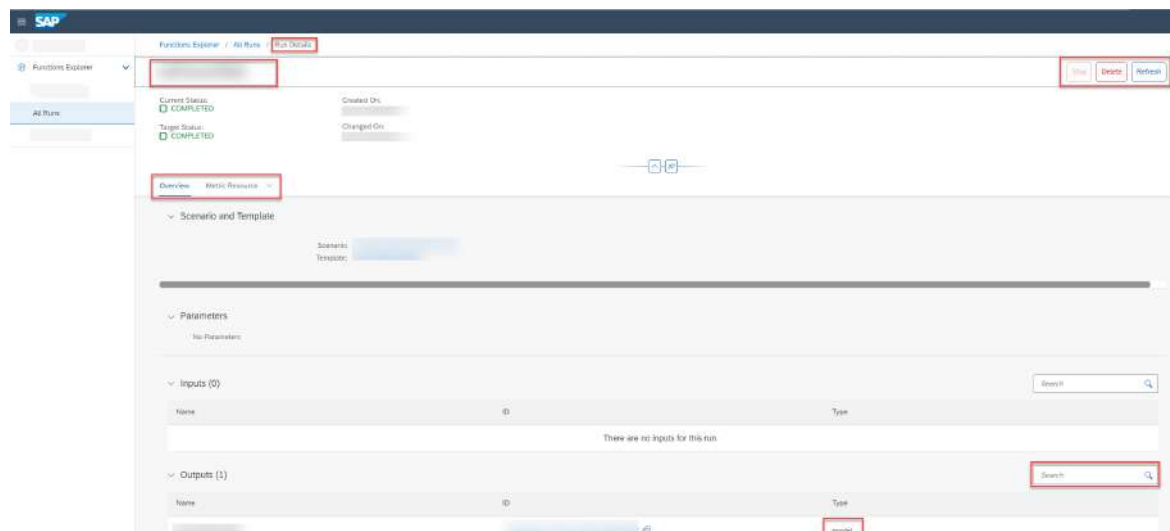
5. **Optional:** Check the *Overview* tab for details such as scenario, template, parameters, input dataset, and output model.
   a. Select the scenario ID to navigate to the scenario details.
   b. Select the template ID to navigate to the template details.
   c. To search for an input or output, enter a value or partial value in the 🔍 *(Search)*field.
6. **Optional:** To investigate the metrics and tags for a run, select the *Metric Resource* tab. See View the Metric Resource for a Run [page 302].
7. **Optional:** To refresh the current status of the run, choose *Refresh*.
8. **Optional:** To stop the run, choose *Stop*. See Stop a Run [page 308].
9. **Optional:** To delete the run, choose *Delete*. See Delete a Run [page 309].

Related Information

Compare Run Metrics [page 305]

6.3.2.3.2 View the Metric Resource for a Run

A metric is the measure of quality (confidence) of a model. During a run, metrics (standard model evaluation metrics and associated labels, tags and custom info) are logged and stored by the run template.

Prerequisites

You have the `scenario_metric_viewer` role or you are assigned to a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

Context

Metrics are reviewed to evaluate the quality of model generated by the run. Metrics are affected by the dataset and parameter values used by the run. A metric resource is a collection of all tracked metrics for a run. You can view a complete metric resource including labels and tags. For more information, see Storing Metric Data.

Procedure

1. Find the run and display its details. For more information, see Investigate a Run [page 300].
2. To view the metric resource, select the *Metric Resource* tab.

   > ⓘ Note
   >
   > The *Metric Resource* tab is only visible if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.
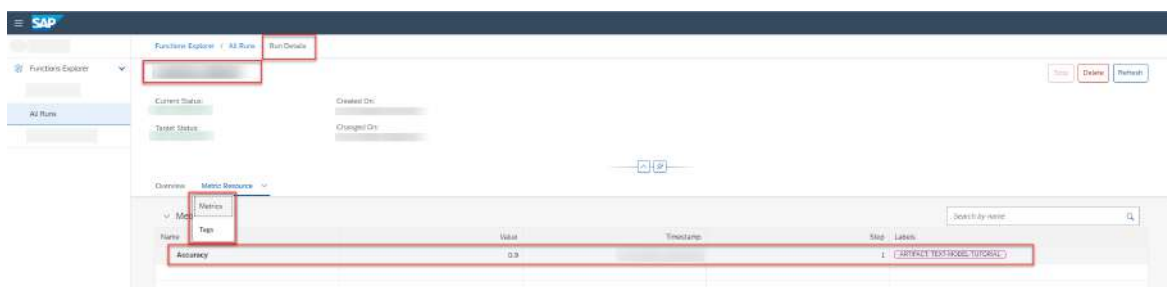
   Metric resource details are displayed, as follows:
   - *Name*
   - *Value*: Quality criteria (model evaluation: Indicates a level of quality, and is dependent on the *Name* (criteria).
   - *Timestamp*: Time at which the metric was logged. Time is adjusted to reflect the local time zone. For metrics that are logged multiple times, the timestamp and step can be used in conjunction to check how a metric has progressed during the run (training process).

- *Step*: Used to uniquely identify or differentiate the results. For example, a model trains iteratively on the same dataset in a single training process. Also known as an epoch.
- *Labels*: Custom information associated with the metric.

3. To display the tags associated with the run, choose *Tags* from the *Metric Resource* tab options, or scroll down the screen.: Quality criteria (model evaluation metrics). For example, accuracy or mean absolute error (MSE).

Related Information

Compare Run Metrics [page 305]

6.3.2.3.3 Compare Runs

You can compare runs to determine which configuration parameters result in optimum results.

You can compare up to five runs, either by comparing metric data or by creating and analyzing charts.

- Metrics provide data about the quality (confidence) of a model. Model quality is affected by the dataset used in the run (training process) and by the input configuration parameters.
  When you compare metrics for runs, multiple run metrics are compared against input configuration parameters. You compare metric data to determine whether further adjustments are required for configurations, or to identify a configuration which is producing optimum results.
- Charts provide a visual representation of a run and resulting model quality. You can choose the chart settings and chart type to graphically compare runs. Chart types include line, bar, column, heat map, and scattered charts. The chart types available for use depend on your source data and chart settings. You can preview a chart before you add it to your chart view.

> ⓘ Note
>
> You can only make comparisons if the metrics capabilities extension in implemented in your AI runtime. For more information, see API Runtime Implementations.

6.3.2.3.3.1 Create Chart to Compare Runs

You can create charts for runs to visually compare quality criteria and values.

Context

You can create multiple charts and view them in the *Chart View*.

> ⓘ Note
>
> Each chart can include data for up to five runs.

Procedure

1. In the *Functions Explorer* app, choose *All Runs*.

2. **Optional:** If you have a large list of runs, you can filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

    a. Enter the run ID or select a status.

    b. Choose *Apply* to apply the filter to the list.

3. Select the runs for comparison and choose *View Metrics*.
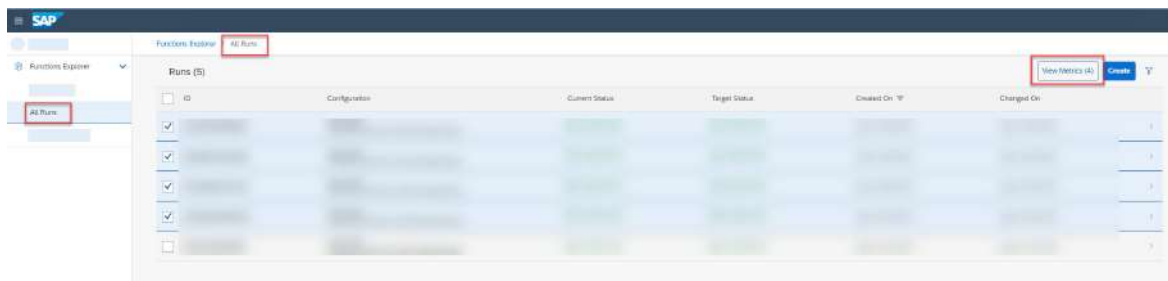
The *Metrics Overview* appears for the selected runs. The run IDs and descriptions are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.
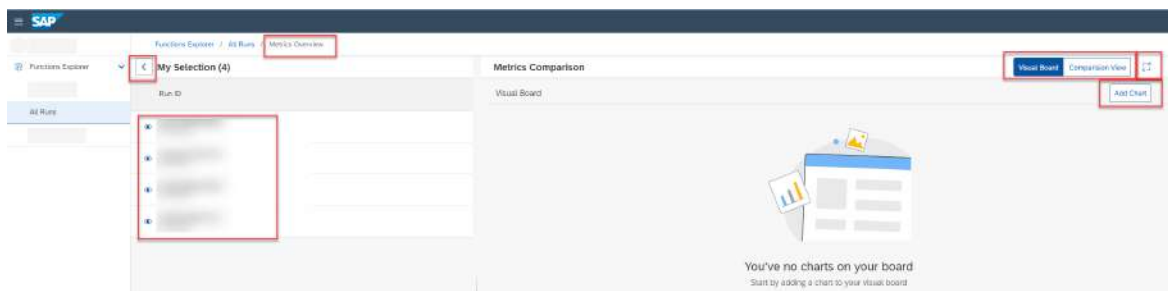
4. Choose *Add Chart* to create a chart based on your selected runs.

    The *Add Chart* dialog appears.

5. Enter the chart settings:

    • Enter a name and description for the chart.

    • In *Chart Settings*, choose `Runs` as the metrics source.

    • In *Comparison Type*, choose your preferred comparison. You can compare metrics to parameters, to the source, or to steps or time. Based on your selection, you'll be prompted to select the metrics and values for comparison.

6. Choose *Preview* to continue to the preview settings.

> ⚠ Caution
>
> If *Preview* is not enabled, review your settings and selections. Some settings are mandatory, and you can't proceed until specified. Some settings and data combinations don't correspond to a valid chart type.

7. In the *Chart Selection* pane, select the chart type (such as column or bar chart). Note, the chart types available depend on the chart settings you defined.

8. Choose *Data Selection* from the dropdown menu to confirm the runs selected for the chart.

    You can show or hide runs from the data selection, and see the impact on the preview chart.

9. Choose *OK* to create the chart. The chart appears in your chart view.

10. **Optional:** Check the chart. Note, if you have multiple charts, you may need to scroll.

    a. To display a chart in full-screen mode, choose ⛶ *(Open Full Screen)*.
    b. To edit a chart, choose ✏ *(Edit)*.
    c. To delete a chart from your chart view choose ✏ *(Delete)*.

6.3.2.3.3.2 Compare Run Metrics

You compare metrics for runs to determine which configuration parameters result in optimum results.

Context

> ⓘ Note
>
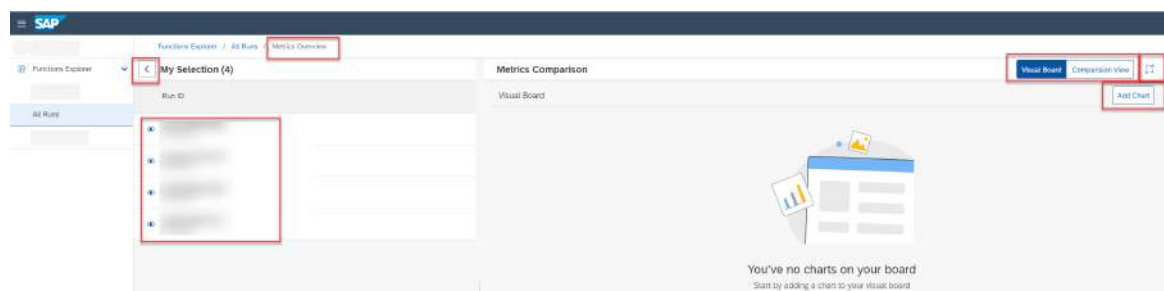> You can compare metrics for up to five runs.

Procedure

1. In the *Functions Explorer* app, choose *All Runs*.

2. **Optional:** If you have a large list of runs, you can filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

    a. Enter the run ID or select a status.
    b. Choose *Apply* to apply the filter to the list.

3. Select the runs for comparison and choose *View Metrics*.

The *Metrics Overview* appears for the selected runs. The run IDs and descriptions are listed in the *My Selection* pane. The *Metrics Comparison* pane defaults to the chart view.

4. Choose *Comparison View* to compare metric data.
5. Investigate the metric data for the selected runs.

- To show the data comparison in full-screen mode, choose ⛶ *(Fullscreen)*.
- To show or hide the runs selected for comparison, choose 👁 *(Show)* or 👁̸ *(Hide)*.
- To show or hide different criteria and comparison data, choose *Configurations*, *Runs*, or *Models*.

Related Information

[Create Chart to Compare Runs \[page 303\]](#)

6.3.2.3.4 Create a Run

A run is a training process which generates a model.

Prerequisites

You have the `mlfunctions_editor` role, or you have been assigned a role collection that contains this role.

For more information, see [Roles and Authorizations \[page 321\]](#).

Context

Runs are used to train a model.

Procedure

1. In the *Functions Explorer* app, choose *All Runs*.

   The *All Runs* screen appears listing all existing runs. Runs are listed by ID, and with additional details such as configuration name and ID, current and target status, created on timestamp, and changed on timestamp.
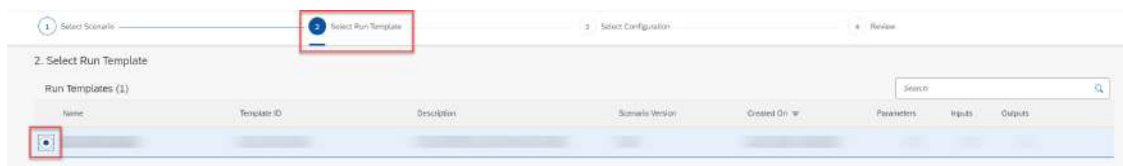
2. Choose *Create* to create a new run.

   The *Create New Run* wizard appears. This wizard has four steps.

3. Select the required data for the new run.

   1. In the *Select Scenario* step, select the scenario from the list and choose *Next*.
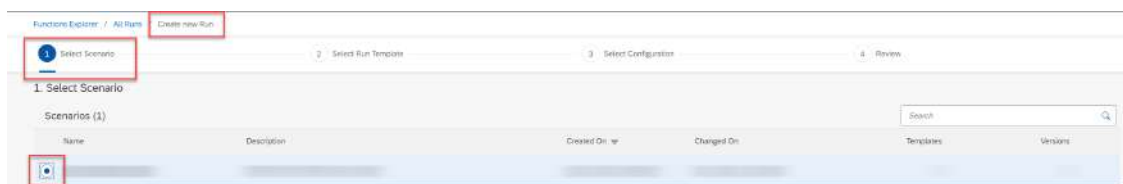
   2. In the *Select Run Template* step, select the run template from the list and choose *Next*.

   3. In the *Select Configuration* step, select the required configuration. The details for the selected configuration are displayed in the right pane.
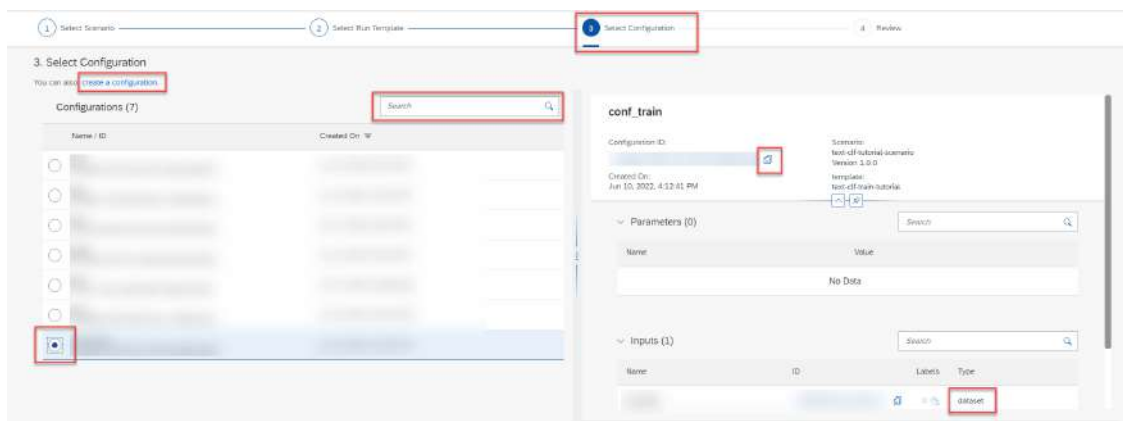
   > → Tip
   >
   > If there is no configuration which matches your data requirements, you can choose *create a configuration*. You will be redirected to create a configuration, and the run you have started will be lost. When you have saved the new configuration, you can re-create the run with the new configuration.

   Confirm the selection and choose *Review*.

   4. In the *Review* step, review the data that you've selected for the new run. Choose *Create* to create the run.

The new run is created and is now displayed in the *All Runs* screen.

6.3.2.3.5   Stop a Run

You can stop a run (training process) and release any computing resource acquired the run.

Prerequisites

You have the role `scenario_job_editor`, or you are assigned to a role collection that contains this role. For more information, see Roles and Authorizations [page 321].

Context

You can only stop a run if it has a *Running* or *Pending* status, which means that computing resources have been allocated to the run. For runs in other statuses, the *Stop* button is not enabled.

Procedure

1. Find the run and display its details. For more information, see Investigate a Run [page 300].
2. Choose *Stop* in the run's header.

3. In the warning dialog, choose *Stop* to confirm the stop.

Results

After a run is stopped, the status of the run changes, and any computing resources are released.

6.3.2.3.6 Delete a Run

You can delete a run if it is no longer required. Deleting a run does not delete the objects associated with the run, such as the model, dataset, configuration, or run template.
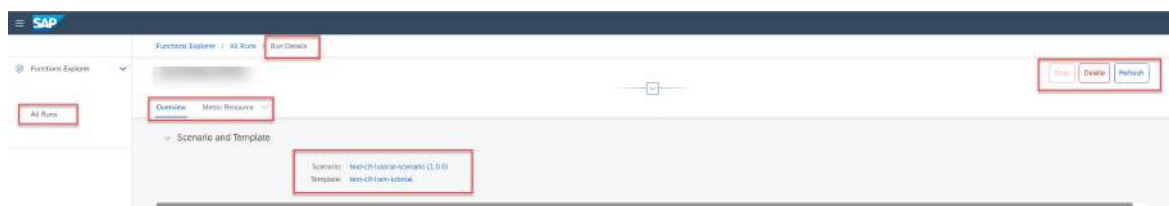
Prerequisites

You have the `scenario_job_editor` role or you are assigned to a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Context

You can delete a run if it has the status *Completed*, *Dead*, *Stopped* or *Unknown*. For runs in other statuses, the *Delete* button is not enabled.

Procedure

1. Find the run and display its details. For more information, see Investigate a Run [page 300].
2. Choose *Delete* in the run's header.
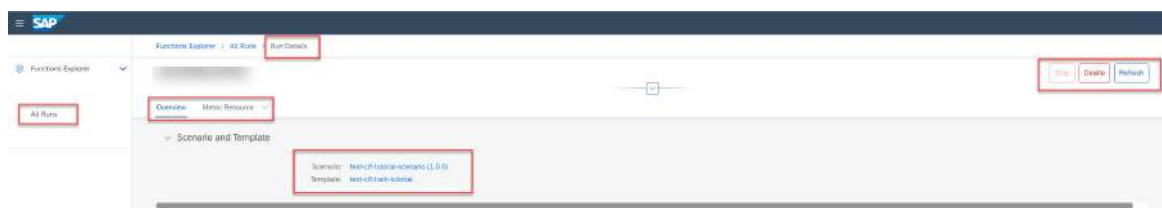
3. In the warning dialog box, choose *Delete* to confirm the deletion.

Results

After a run is deleted, you are redirected to the *All Runs* page. The deleted run is no longer available.

Any model generated by the run is not impacted by the deletion.

6.3.2.4　Deployments

You use deployments to make online predictions.

You use SAP AI Launchpad to create deployments for your runtime connection. Deployments that are implemented on an SAP AI Core runtime produce `HTTPS` endpoints.

About Deployments

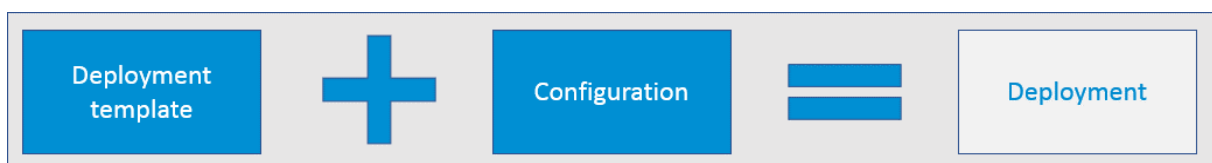A deployment is used to make a prediction. A deployment creates a model server and generates a deployment URL. The URL is used for inference.

- A deployment template defines the expected parameters and input dataset that is required for the deployment process.
- Values for parameters and input dataset are provided by a configuration. A deployment takes one or more models and parameters from a configuration.
- A deployment is created to start the deployment process with these values.
- A deployment generates a URL for making online predictions.

Deployment States

When a deployment has a *running* status, you can use it to make predictions. A deployment with a *stopped* status cannot be reused. To run a deployment again, create another deployment using the same configuration, and retain the combination of values for the deployment template and model. Multiple configurations can be created for a deployment. However, a *running* deployment can only refer to one configuration. You can update a deployment to change the referenced configuration.

Deployments can have any of the following statuses:

Pending
Running
Stopping
Stopped
Dead
Unknown

The following figure shows how deployment statuses change following the initial status of *Pending*:
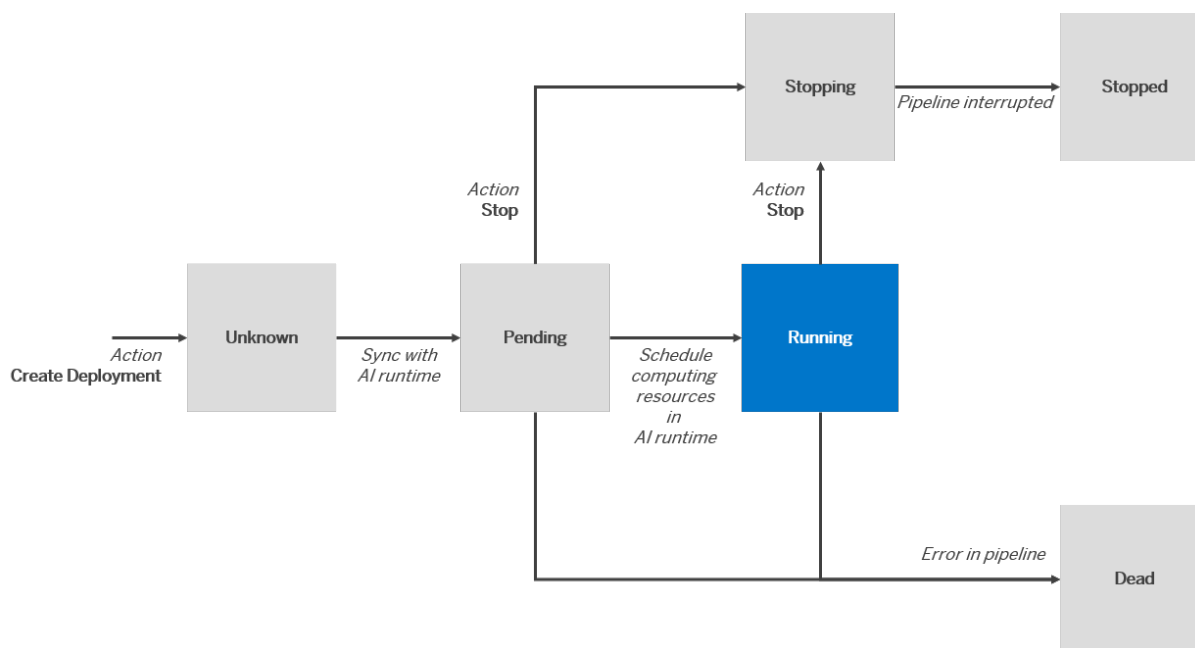
Deployment Status Flow

When a deployment leaves *Running* status, the computing resources that were used by the AI runtime are released.

6.3.2.4.1 Investigate a Deployment

You can explore deployment details for information about the deployment process for a model.

Prerequisites

You have the `mlfunctions_viewer` or `mlfunctions_editor` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

A deployment's details consists of scenario and template, parameters, and input model. The status of the deployment determines what you can do with the deployment. For example, a deployment with a running status can be stopped but not updated.

Procedure

1. In the *Functions Explorer* app, choose *All Deployments*.

   The *All Deployments* screen appears listing all existing deployments. Deployments are listed by ID, and with additional details such as configuration name and ID, current and target status, created on and changed on timestamps.

2. **Optional:** Filter the list by choosing ▽ *(Filter)*. The *Filter* dialog appears.

   a. Enter the deployment ID or status.
   b. Choose *Apply* to apply the filter to the list.

3. **Optional:** To create a new deployment, choose *Create*. See Create a Deployment [page 313].

4. To view the details for an individual deployment, select the deployment in the list or choose ❯ *(More)*.
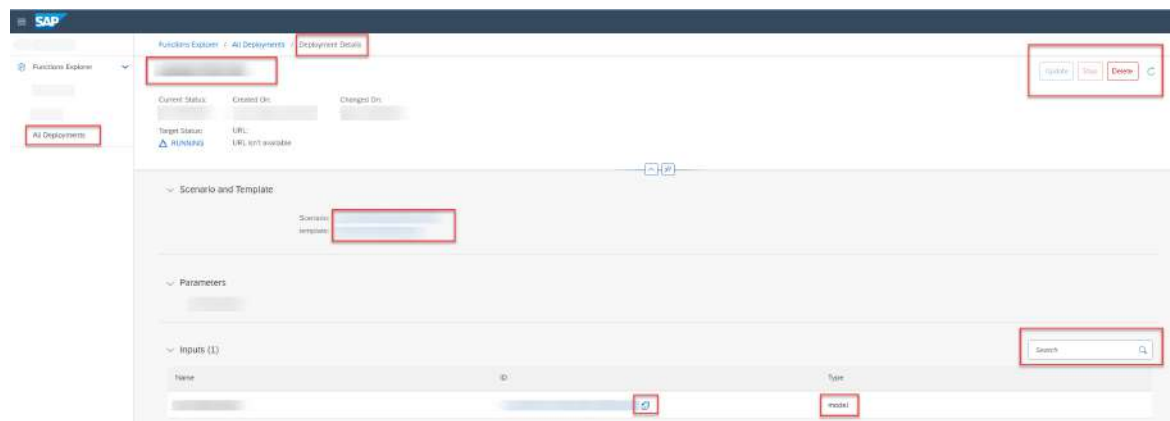
   The *Deployment Details* screen appears with details such as the current and target status, scenario, deployment template, parameters, and dataset.

   > → Tip
   >
   > Timestamp details for each step in the deployment process can be seen in the header. These dates and times show when the deployment was created, submitted, started, and finished. The process duration is also displayed. Timestamps are displayed in your local time zone.

5. **Optional:** Check the deployment details.

   a. Select the scenario ID to navigate to the scenario details.
   b. Select the template ID to navigate to the template details.
   c. To search for an input or a parameter, enter a value or partial value in the 🔍 *(Search)* field.
   d. To copy the ID of the model, choose 🗐 *(Copy)*.

6. **Optional:** To refresh the current status of the deployment, choose ⟳ *(Refresh)*.

7. **Optional:** To update the deployment with a different configuration, choose *Update*. See Update a Deployment [page 314].

8. **Optional:** To stop the deployment, choose *Stop*. See Stop a Deployment [page 315].

9. **Optional:** To delete the deployment, choose *Delete*. See Delete a Deployment [page 316].

6.3.2.4.2 Create a Deployment

A deployment uses a model and data to make a prediction.

Prerequisites

You have the `mlfunctions_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Context

Deployments are used to make online predictions.

Procedure

1. In the *Functions Explorer* app, choose *All Deployments*.

   The *All Deployments* screen appears listing all existing deployments. Deployments are listed by ID, and with additional details such as configuration name and ID, current and target status, created on timestamp, and changed on timestamp.

2. Choose *Create* to create a new deployment.

   The *Create New Deployment* wizard appears. This wizard has four steps.

3. Select the required data for the new deployment.
   1. In the *Select Scenario* step, select the scenario from the list and choose *Next*.

   2. In the *Select Deployment Template* step, select the deployment template from the list and choose *Next*.
   3. In the *Select Configuration* step, select the required configuration. The details for the selected configuration are displayed in the right pane.

Confirm the selection and choose *Review*.
4. In the *Review* step, review the data that you've selected for the new deployment. Choose *Create* to create the deployment.

The new deployment is created and is now displayed in the *All Deployments* screen.

6.3.2.4.3   Update a Deployment

You can update a deployment with your choice of configuration.

Prerequisites

You have the `scenario_deployment_viewer` or `scenario_deployment_editor` role, or you have been assigned a role collection that contains one of these roles.

For more information, see Roles and Authorizations [page 321].

Context

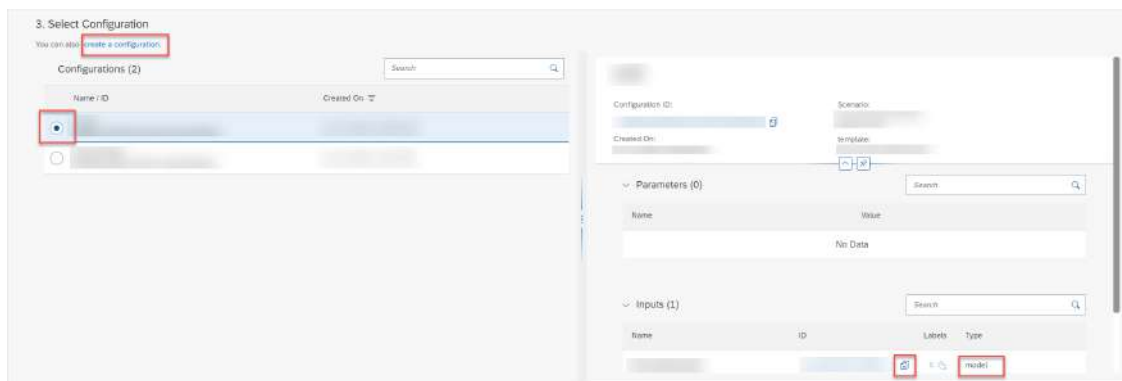Multiple configurations can be associated with a deployment. However, a deployment can only use one configuration. When you update a deployment, you can choose to apply a new configuration, or re-use an older or previously used configuration. For example:

- Your existing deployment serves models A and B, but now you'd like to replace model A with a new model C. You want to use the same deployment and perform inference using the existing deployment URL endpoint.
- You want to update an existing deployment with new parameter bindings, such as number of replicas, autoscaling annotation, or ENV variables.

Deployments can only be updated if their status is *pending*, *running* or *dead*.

The updated deployment retains the inference URL. If you use a new deployment configuration, inference requests continue to work.

Procedure

1. Find the deployment and display its details. See Investigate a Deployment [page 311].
2. Choose *Update* to display the *Update Deployment* dialog. All available configurations are listed. You can display summary details for each configuration, and use this information to compare details such as parameters and input artifacts.

3. Select the required configuration from the list, and choose *Update* to update the deployment.

   > → Tip
   >
   > The system indicates the configuration which is already in use. You cannot update the deployment with this configuration.

4. Check the *Deployment Details* screen, and confirm that the deployment is *running* with the selected configuration.

   > ⓘ Note
   >
   > If the update was not effective and the deployment status is *dead*, you'll receive an error message. The error message contains the configuration ID of the last configuration that resulted in a *running* deployment. You can update the deployment with this last configuration to effectively undo or rollback the error, and return the deployment to a *running* status.

6.3.2.4.4   Stop a Deployment

You can stop a deployment and release any computing resources acquired for the deployment.

Prerequisites

You have the `mlfunctions_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Context

You can stop a deployment if it has the status *Running* or *Pending*, which means that computing resources have been allocated for the deployment. For deployments in other statuses, the *Stop* button is not enabled.

Procedure

1. Find the deployment and display its details. See Investigate a Deployment [page 311].
2. Choose *Stop* in the deployment's header.
3. In the *Warning* dialog, choose *Stop*.

Results

The status of the deployment is now stopped, and any computing resources acquired for the deployment are released.

6.3.2.4.5 Delete a Deployment

You can delete a deployment if it is no longer required. Deletion removes information about a deployment but does not delete the objects involved.

Prerequisites

You have the `mlfunctions_editor` role, or you have been assigned a role collection that contains this role.

For more information, see Roles and Authorizations [page 321].

Context

You can delete a deployment if it has the status *Dead*, *Stopped*, or *Unknown*. For deployments in other statuses, the *Delete* button is not enabled.

Procedure

1. Find the deployment and display its details. See Investigate a Deployment [page 311].
2. Choose *Delete* in the deployment's header.
3. In the *Warning* dialog, choose *Delete* to confirm the deletion.

Results

After the deployment has been deleted, all its information is lost. You will not be able to access it again.

7 Example End-to-End Use Case

To become familiar with the fundamentals of using SAP AI Core and SAP AI Launchpad, you can refer to the developer tutorials.

Getting Started

The developer tutorials demonstrate how you can use SAP AI Launchpad together with an AI API connection to SAP AI Core. Before you start to work through the tutorials, you must set up your SAP AI Core instance. For more information, see the tutorials available within Create Your First Machine Learning Project Using SAP AI Core .

> → Recommendation
>
> An assumption throughout the developer tutorials is that you use SAP AI Core as the AI runtime. SAP AI Core is not included in the subscription of SAP AI Launchpad. It is a separate product and requires a separate subscription.

8   Security

Here, we'll explain some of the security aspects of SAP AI Launchpad.

8.1   Data Protection and Privacy

For general information about data protection and privacy on SAP Business Technology Platform, see Data Protection and Privacy.

Data protection is associated with numerous legal requirements and privacy concerns. In addition to compliance with general data protection and privacy acts, it is necessary to consider compliance with industry-specific legislation in different countries. SAP provides specific features and functions to support compliance with regard to relevant legal requirements, including data protection. SAP does not give any advice on whether these features and functions are the best method to support company, industry, regional, or country-specific requirements. Furthermore, this information should not be taken as advice or a recommendation regarding additional features that would be required in specific IT environments. Decisions related to data protection must be made on a case-by-case basis, taking into consideration the given system landscape and the applicable legal requirements.

> ⓘ Note
>
> SAP does not provide legal advice in any form. SAP software supports data protection compliance by providing security features and specific data protection-relevant functions, such as simplified blocking and deletion of personal data. In many cases, compliance with applicable data protection and privacy laws will not be covered by a product feature. Definitions and other terms used in this document are not taken from a particular legal source.

SAP Data Protection and Privacy (DPP) requirements and the European Union's General Data Protection Regulations (GDPR) mandate the protection of personal and private data at higher standards than other customer data. SAP machine learning services do not import, output, or process any structured personal or private customer data, and cannot distinguish personal or private data from other types of data. Customers are therefore obliged to fulfill the GDPR personal data broker obligations if such data is present.

> ⚠ Caution
>
> Don't store personal data in prompts when using the generative AI hub. Personal data is any data that can be used alone, or in combination, to identify the person that the data refers to.

Read Access Logging

SAP AI Launchpad stores tenant id (identity-zone, JWT token's zid) in application and audit logs, which are sent to the ELK stack and audit log service respectively. SAP Business Technology Platform persists application logs in the ELK log-stash for 7 days.

Deletion

The audit log data stored for your account will be retained for 30 days, after which it will be deleted.

For users who are manually offboarded from the generative AI hub data deletion is triggered immediately.

For data deletion resulting from the deletion of a resource group, deletion is triggered within 24 hours.

For data deletion from the generative AI hub resulting from the removal of an SAP AI Core tenant, SAP AI Launchpad connection to SAP AI Core or SAP AI Launchpad instance, data will be retained for 30 days, after which it will be deleted.

> ⓘ Note
>
> User data is saved in one region only and can only be retrieved or deleted by an instance of AI launchpad in that region.

8.2   Security and Customer Data Protection

SAP product standard security and the data protection and privacy (DPP) requirements set high standards and obligations when it comes to securing and protecting customer data that is entrusted to SAP.

Customer data protection is handled in three ways:

- Customer data is imported, output, and processed by the services for no purpose beyond that to which the customer has subscribed.
- Customer data is protected from malicious access by security technologies that include authentication and authorization.
- Customer data is protected from accidental exposure to SAP administrators or support persons by security policies, access controls, and monitoring.

8.3   Encryption in Transit

Communication with the service, including data upload and download, is encrypted using the transport layer security (TLS) protocol. SAP services support only the latest protocol versions (that is, TLS v1.2 and later) and strong cipher suites. Your systems must use the supported protocol versions and cipher suites to set up secure

communication with the services. They must also validate the certificates against the services' domain names to avoid man-in-the-middle attacks.

8.4 User Authentication and Authorization

SAP machine learning services use the advanced OAuth 2.0 protocol for authentication and authorization. Customer systems must implement strong protection to safeguard the authentication credentials (client credentials).

SAP AI Launchpad uses standard SAP Authorization and Trust Management service services, which lets you manage user authorizations and trust to identity providers.

SAP AI Launchpad integrates with the User Account and Authentication service (SAP CP XSUAA). This allows the federation of external and custom identity providers (IdP).

Configuring External Identity Providers

Information on how to configure external identity providers is available in the SAP Business Technology Platform documentation:

- For information about the configuration of identity providers via UAA, see Data Protection and Privacy.
- For information about establishing trust between UAA and the Identity Authentication service (IAS), see Establish Trust and Federation Between UAA and Identity Authentication.

The configuration is handled within the security configuration dashboard of the subaccount that contains the relevant instance.

8.4.1 Advanced Role Management

8.4.1.1 Roles and Authorizations

SAP AI Launchpad provides default role collections that you can assign to users. The role collections determine which actions a user is able to carry out in SAP AI Launchpad. You can also create your own role collections and assign the required roles to them.

SAP AI Launchpad provides standard or default role collections. The default role collections bundle the roles required for likely SAP AI Launchpad tasks and app functions. However, if these role collections do not suit your organizational needs, you can create your own role collections.

Default Role Collections

Relevance	Default Role Collection	Description	Includes Roles
SAP AI Launchpad	`ailaunchpad_connections_editor`	Provides roles to view, create, edit, and delete connections to your AI runtime (for example, SAP AI Core)	`viewer` `connections_editor`
SAP AI Launchpad	`ailaunchpad_allow_all_resourcegroups`	Provides access to all resource groups	`allow_all_resourcegroups`
ML Operations	`ailaunchpad_mloperations_viewer`	Provides roles to view all contents of scenarios and resource groups	`viewer` `mloperations_viewer`
ML Operations	`ailaunchpad_mloperations_editor`	Provides roles to view all contents of scenarios, and to view and edit contents of resource groups	`viewer` `mloperations_editor`
SAP AI Core Administration	`ailaunchpad_aicore_admin_viewer`	Provides roles to view authentications required for AI workflows involving SAP AI Core (AI runtime)	`viewer` `aicore_admin_viewer_all`
SAP AI Core Administration	`ailaunchpad_aicore_admin_editor`	Provides roles to edit authentications required for AI workflows involving SAP AI Core (AI runtime)	`viewer` `aicore_admin_editor_all`
Functions Explorer	`ailaunchpad_functions_explorer_viewer_v2`	Provides roles to view scenarios and all ML resources of a scenario	`viewer` `connections_viewer` `functions_explorer` `mlfunctions_viewer` `scenario_executable_viewer` `scenario_metadata_viewer` `scenario_configuration_viewer` `scenario_job_viewer` `scenario_artifact_viewer` `scenario_metric_viewer`

Relevance	Default Role Collection	Description	Includes Roles

Functions Explorer	`ailaunchpad_functions_explorer_editor_v2`	Edit scenarios and all ML resources of a scenario	`viewer`
			
			`connections_viewer`
			
			`functions_explorer`
			
			`mlfunctions_editor`
			
			`scenario_executable_viewer`
			
			`scenario_metadata_viewer`
			
			`scenario_metric_viewer`
			
			`scenario_configuration_editor`
			
			`scenario_job_editor`
			
			`scenario_artifact_editor`

SAP AI Launchpad	`ailaunchpad_connections_editor_without_genai`	Provides roles to view, create, edit, and delete connections to your AI runtime (for example, SAP AI Core) without generative AI hub	`viewer_without_genai`
			
			`connections_editor`

ML Operations	`ailaunchpad_mloperations_viewer_without_genai`	Provides roles to view all contents of scenarios and resource groups without generative AI hub	`viewer_without_genai`
			
			`mloperations_viewer`

ML Operations	`ailaunchpad_mloperations_editor_without_genai`	Provides roles to view and edit all contents of scenarios, and to view and edit contents of resource groups without generative AI hub	`viewer_without_genai`
			
			`mloperations_editor`

Relevance	Default Role Collection	Description	Includes Roles

Functions Explorer	`ailaunchpad_functions_explorer_viewer_v2_without_genai`	Provides roles to view scenarios and all ML resources of a scenario without generative AI hub	`viewer_without_genai`
			
			`connections_viewer`
			
			`functions_explorer`
			
			`mlfunctions_viewer`
			
			`scenario_executable_viewer`
			
			`scenario_metadata_viewer`
			
			`scenario_configuration_viewer`
			
			`scenario_job_viewer`
			
			`scenario_artifact_viewer`
			
			`scenario_metric_viewer`

Functions Explorer	`ailaunchpad_functions_explorer_editor_v2_without_genai`	Edit scenarios and all ML resources of a scenario without generative AI hub	`viewer_without_genai`
			
			`connections_viewer`
			
			`functions_explorer`
			
			`mlfunctions_editor`
			
			`scenario_executable_viewer`
			
			`scenario_metadata_viewer`
			
			`scenario_metric_viewer`
			
			`scenario_configuration_editor`
			
			`scenario_job_editor`
			
			`scenario_artifact_editor`

SAP AI Core Administration	`ailaunchpad_aicore_admin_viewer_without_genai`	Administrator (SAP AI Core) viewer, without access to generative AI hub	`viewer_without_genai`
			
			`aicore_admin_viewer_all`

SAP AI Core Administration	`ailaunchpad_aicore_admin_editor_without_genai`	Administrator (SAP AI Core) editor, without access to generative AI hub	`viewer_without_genai`
			
			`aicore_admin_editor_all`

Relevance	Default Role Collection	Description	Includes Roles

Generative AI Hub	`ailaunchpad_genai_experimenter`	Prompt experimentation, viewing disclaimers and orchestration experimentation in generative AI hub.	`viewer_without_genai`
			
			`genai_experimenter`

Generative AI Hub	`ailaunchpad_genai_manager`	Prompt experimentation, managing prompts, viewing disclaimers and orchestration experimentation in generative AI hub.	`viewer_without_genai`
			
			`genai_experimenter`
			
			`genai_manager`

Generative AI Hub	`ailaunchpad_genai_administrator`	Administration user data deletion and reading, writing and deletion of disclaimers in generative AI hub.	`viewer_without_genai`
			
			`genai_administrator`

If you want to create your own role collections, you can assign roles to them based on the following table.

Default Roles

Role	Allows Users To
`allow_connections`	Manage custom access to runtime connections in SAP AI Launchpad
`connections_viewer`	View runtime connections in SAP AI Launchpad
`connections_editor`	Edit runtime connections in SAP AI Launchpad
`viewer`	Access SAP AI Launchpad (default role for all users)
`aicore_admin_repositories_viewer`	View Git repositories in the *SAP AI Core Administration* app
`aicore_admin_repositories_editor`	Add, edit, or remove Git repositories in the *SAP AI Core Administration* app
`aicore_admin_applications_viewer`	View applications in the *SAP AI Core Administration* app
`aicore_admin_applications_editor`	Create, edit, or delete applications in the *SAP AI Core Administration* app
`aicore_admin_dockerregistrysecret_viewer`	View Docker registry secrets in the *SAP AI Core Administration* app
`aicore_admin_dockerregistrysecret_editor`	Add, edit, or remove Docker registry secrets in the *SAP AI Core Administration* app
`aicore_admin_resourcegroup_viewer`	View resource groups in the *SAP AI Core Administration* app
`aicore_admin_resourcegroup_editor`	Create, edit, or delete resource groups in the *SAP AI Core Administration* app
`aicore_admin_objectstoresecret_viewer`	View object store secrets in the *SAP AI Core Administration* app
`aicore_admin_objectstoresecret_editor`	Add, edit, or remove object store secrets in the *SAP AI Core Administration* app

Role	Allows Users To
scenario_metadata_viewer	View scenarios and scenario versions
scenario_executable_viewer	View executables of a scenario
scenario_configuration_viewer	View configurations of a scenario
scenario_configuration_editor	Edit configurations of a scenario
scenario_deployment_viewer	View deployments of a scenario
scenario_deployment_editor	Edit deployments of a scenario
scenario_deployment_predictor	Invoke deployments of a scenario
scenario_execution_viewer	View executions of a scenario
scenario_execution_editor	Edit executions of a scenario
scenario_artifact_viewer	View artifacts of a scenario
scenario_artifact_editor	Edit artifacts of a scenario
scenario_metric_viewer	View tracking metrics of an execution
resourcegroup_viewer	View resource groups
allow_all_resourcegroups	Template for allowing access to all resource groups for a connection
functions_explorer	Template for viewing functions explorer for SAP AI Launchpad
operations_manager	Template for viewing operations manager for SAP AI Launchpad
scenario_job_viewer	View jobs of a scenario
scenario_job_editor	Edit jobs of a scenario
mloperations_viewer	Viewer role for *ML Operations* app
mloperations_editor	Editor role for *ML Operations* app
artifact_register	Register artifacts in *ML Operations* app
mlfunctions_viewer	Viewer role *Functions Explorer* app
mlfunctions_editor	Editor role *Functions Explorer* app
aicore_admin_viewer_all	Viewer role *SAP AI Core Administration* app

Role	Allows Users To
`aicore_admin_editor_all`	Editor role *SAP AI Core Administration* app
`execution_schedules_editor`	Edit execution schedules
`execution_schedules_viewer`	View execution schedules
`aicore_admin_genericsecret_editor`	Create, edit or delete generic secrets in the SAP AI Core runtime through the *SAP AI Core Administration* app
`aicore_admin_genericsecret_viewer`	View generic secrets in the SAP AI Core runtime through the *SAP AI Core Administration* app
`genai_experimenter`	Run prompts in the `generative AI hub prompt editor` Build and test orchestration workflows in the `generative AI hub orchestration service` Explore available models in the generative AI hub and make an informed model selection using the model library
`genai_manager`	Create, update and run prompts in the `generative AI hub prompt editor` Read and delete your saved prompts in the `generative AI hub` prompt manager Build, test, and manage orchestration workflows in the `generative AI hub orchestration service` Explore available models in the generative AI hub and make an informed model selection using the model library
`genai_administrator`	Delete user data in the **generative AI hub** Explore available models in the generative AI hub and make an informed model selection using the model library
`prompt_experimenter`	Run prompts in the `generative AI hub prompt editor`. Build and test orchestration workflows in the `generative AI hub orchestration service`
`prompt_manager`	Create, update and run prompts in the `generative AI hub prompt editor`, read and deleted your saved prompts in the `generative AI hub` prompt manager
`prompt_administrator`	Delete user data in the generative AI hub. Create, edit or delete disclaimers in the generative AI hub.
`viewer_without_genai`	Access SAP AI Launchpad, default role for users, without generative AI hub
`orchestration_executor`	Build and test orchestration workflows in the `generative AI hub orchestration service` Explore available models in generative AI hub and make an informed model selection using the model library

Role	Allows Users To
`prompt_media_executor`	Upload images in the `generative AI hub chat and prompt editor`.
`grounding_manager`	View, create and delete document repositories and pipelines in the grounding management app in generative AI hub.
`grounding_viewer`	View document repositories and pipelines in the grounding management app in generative AI hub.
`custom_evaluation`	Custom evaluation access in generative AI hub, for creating and viewing custom evaluation jobs and viewing results.
`files_editor`	View, download, create, update, and delete data in registered object stores.in the SAP AI Core runtime.
`files_viewer`	Download data in registered object stores.in the SAP AI Core runtime.
`orchestration_experimenter`	View and run orchestration workflows in the generative AI hub
`orchestration_manager`	View, create and run orchestration workflows in the generative AI hub
`prompt_template_experimenter`	View and run prompt templates in the `generative AI hub`
`prompt_template_manager`	Create and view prompt templates in the `generative AI hub`
`prompt_template_developer`	Manage the lifecycle of prompt templates in the `generative AI hub`

> ⓘ Note
>
> Generative AI hub is available through the default `viewer` role.
>
> Access to the generative AI hub can be revoked by assigning the equivalent role `without_genai`. For example, the `viewer` and `viewer_without_genai` are equivalent roles, with and without generative AI hub capabilities respectively.

Related Information

8.4.1.2 Create Role Collection

You create your own role collections and assign roles to them so that you can further manage users' access to content within SAP AI Launchpad.

Prerequisites

Context

You create custom role collections to manage access settings for the following:

- *Workspaces* app users, so that applicable resource groups are available to nominated users
- *Workspaces* app users, so that applicable connections are available to nominated users
- SAP Support users, so that they have custom access to available connections in your organization

Procedure

1. In SAP BTP cockpit, navigate to your subaccount and choose *Role Collections*.
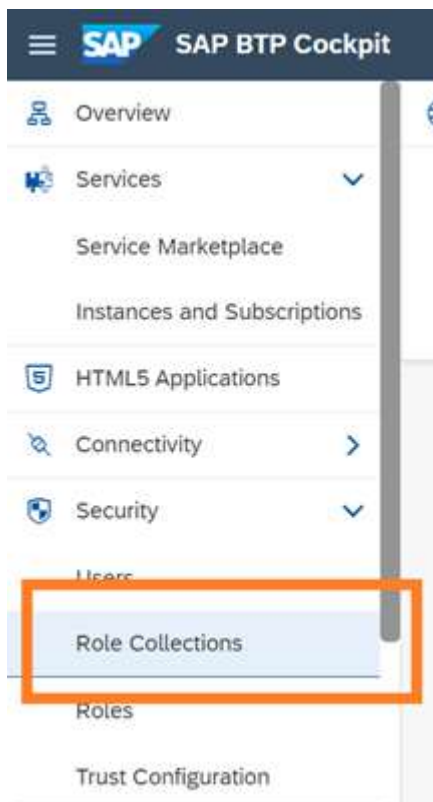
2. Choose ✛ *(Add)* to add a role collection.

> → Remember
>
> A role collection groups roles and users (and their email IDs) with a shared persona.
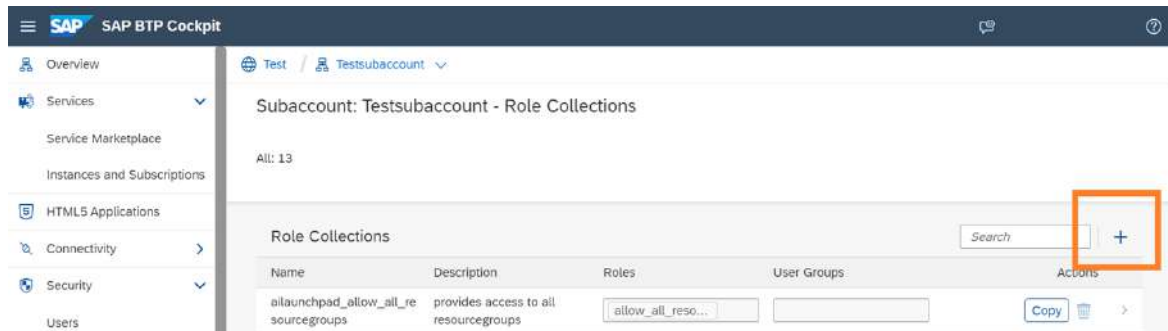
3. Enter a name and description for the group of target users and choose *Create*.

4. Choose *Edit* to add nominated users to this role collection.

5. Enter the email ID of the nominated user and choose *Save*.

   If required, you can add more users (and their emails IDs) by choosing ✛ *(Add)*.
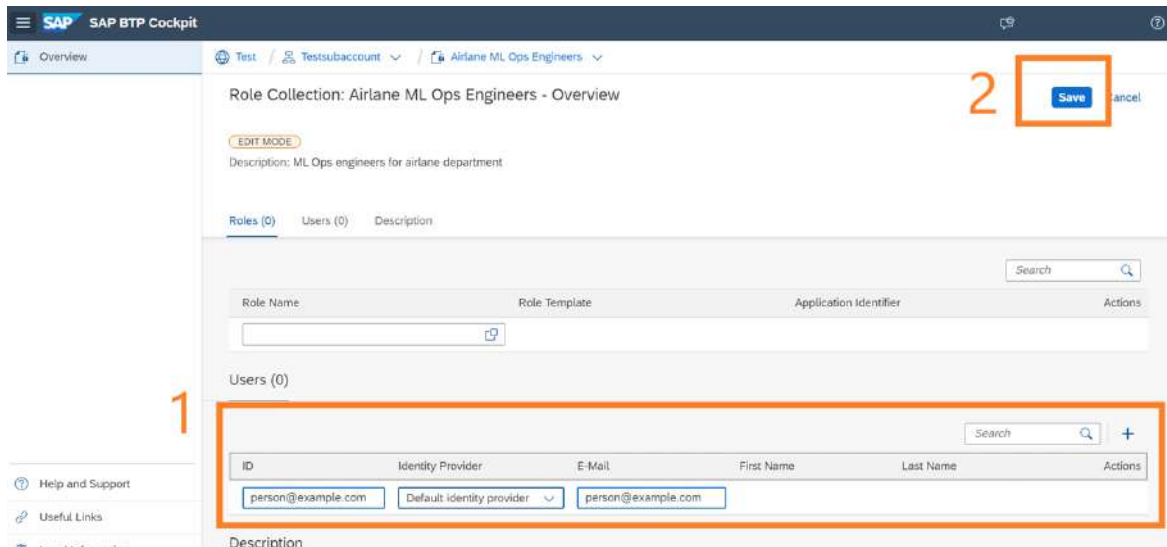
6. Choose *Save* to save the role collection.

    The role collection is now listed in the SAP BTP Cockpit and is available for assignment. For example, you can use the role collection to manage custom access for resource groups or connections. See Custom Access for Connections [page 331] and Custom Access for Resource Groups [page 333].

8.4.1.3   Custom Access for Connections

You can selectively control users' access to connections within SAP AI Launchpad.

Prerequisites

You have created a role collection for use with custom connections. See Create Role Collection [page 329].

Context

A role collection represents a shared user persona. A role collection groups roles and users (and their email IDs) who complete similar tasks.

By default, users of the *Workspaces* app have access to all the connections in their AI runtime. You provide custom access to selectively show connections to nominated roles and users.

> → Tip
>
> For example, you may selectively control access if some connections involve sensitive data, or to simplify the selection options for users. You may also selectively provide access to an SAP Support user. See Custom Access for Support User [page 337].

You manage access to connections by creating a custom role collection within SAP AI Launchpad, and assigning it a role based on the `allow_connections` role template. Nominated users within the *Workspaces* app, are then shown only the connections applicable for their work.

Procedure

1. In SAP BTP cockpit, navigate to your subaccount and choose *Roles* to access the list of roles.
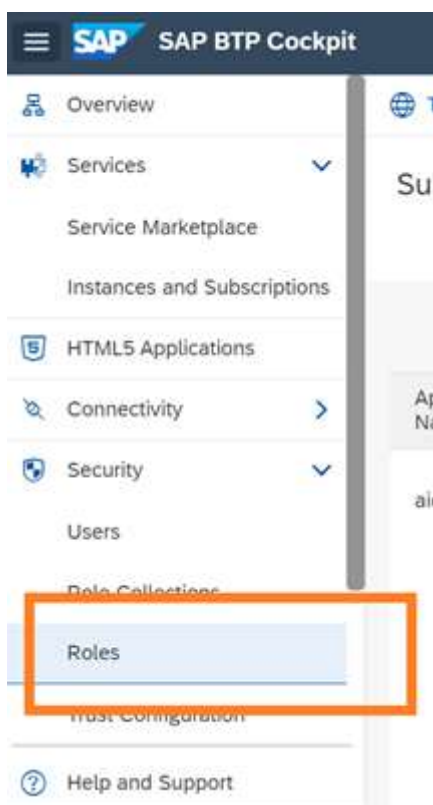
2. Within the `ailaunchpad` application, find the `allow_connections` role template and choose *Create Role*.

3. Complete the role details.

    a. In the *Configure Role* wizard step, enter a name and description for the group of target users. You'll see that the `allow_connections` role template is automatically assigned to the role. Choose *Next* to continue.

    b. In the *Configure Attributes* wizard step, enter the required connections in the *Values* column for the attribute `allow_connections`. Choose *Next* to continue.

    > → Tip
    >
    > To enter multiple connections, press *Enter* between connections and confirm the connection name. If you enter a value of `all`, you override any other connection value entered. This restores the system default and any custom settings are ignored.

    c. In the *Select Role Collections* wizard step, search for and select the custom role collection that you've created for custom connections access. Choose *Next* to continue.

d. In the *Review* wizard step, check the role details and choose *Finish*.

Results

The role is now listed in the SAP BTP cockpit and the custom connection settings are applied. Users assigned to the `allow_connections` role (or a role collection that contains it), now have custom access to connections in their *Workspaces* app.

Related Information

Assign Connection to Workspace [page 164]

8.4.1.4   Custom Access for Resource Groups

You can selectively control users' access to resource groups (within an AI runtime connection) in SAP AI Launchpad.

Prerequisites

You are using the standard role collection, or you have created a role collection for use with custom resource groups. See Create Role Collection [page 329].

The users do not have the role collection `ailaunchpad_allow_all_resourcegroups` assigned.

Context

You can selectively control access to resource groups within an AI runtime connection. Nominated users of the *Workspaces* app, are then shown only those resource groups which are applicable for their work.

Procedure

1. In SAP BTP cockpit, navigate to your subaccount and choose *Roles* to access the list of roles.
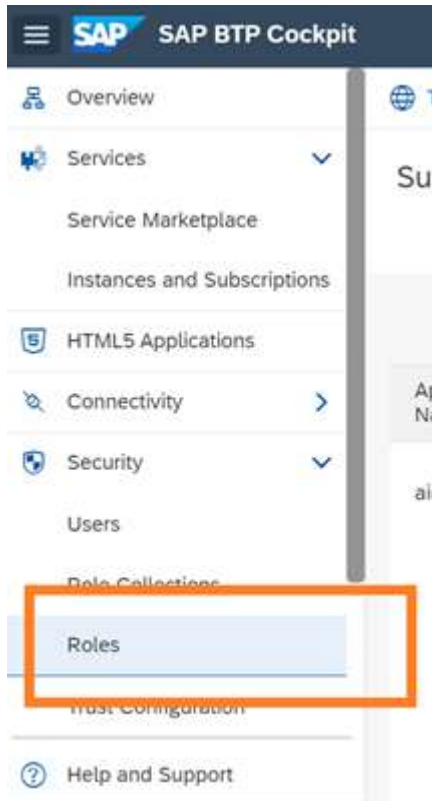
2. Within the `ailaunchpad` application, find the `allow_all_resourcegroups` role template, and choose *Add Using Same Role Template*.

The *Create Role* wizard appears.

3. Complete the role details.

   a. In the *Configure Role* wizard step, enter a name and description for the group of target users. You'll see that the `allow_all_resourcegroups` role template is automatically assigned to the role. Choose *Next* to continue.

b. In the *Configure Attributes* wizard step, enter the required resource group IDs in the *Values* column for the attribute `allowed_all_resourcegroups`. Choose *Next* to continue.

c. In the *Select Role Collections* wizard step, search for and select the custom role collection that you've created for custom access. Choose *Next* to continue.

d. In the *Review* wizard step, check the role details and choose *Finish*.

Results

The role is now listed in the SAP BTP cockpit and the custom resource group settings are applied. Users currently assigned to the `allow_all_resource_groups` role (or a role collection that contains it), now have custom access to resource groups in their *Workspaces* app.

Related Information

8.4.1.5   Custom Access for Support User

An SAP Support user may provide troubleshooting support in case of issues in your production environment.

Context

At some point, you may need to create an incident for an issue that you encounter with SAP AI Launchpad. To investigate the issue, an SAP Support user may require access to your production environment.

SAP Support users access your production environment via an internal user and access management tool called Cloud Access Manager (CAM). CAM, much like the SAP ID Service, allows for the packaging of variable levels of access to your production environment in a single requestable profile. The access request is reviewed by a set of named approvers and either approved or rejected. The profiles also define a time period for which access is granted. CAM will automatically revoke access once the time period has elapsed. Users may request access again, but this will once again require review and approval.

On approval, the SAP support user will be able to access your SAP AI Launchpad instance and provide administrative or operations support.

To provide an SAP Support user with custom access to your connections, see Custom Access for Connections [page 331].

Related Information

8.5   Auditing and Logging Information

Here you can find a list of the security events that are logged by SAP AI Launchpad.

Security Events Written in Audit Logs

What Events Are Logged	How to Identify Related Log Events
Login Successful	Message containing a time stamp, tenant, and user IDs; data containing event details.

What Events Are Logged	How to Identify Related Log Events
Scope check failed	Message containing a time stamp, tenant, and user IDs; data containing "User not authorized" and event details.
Ops Backend	Message containing a time stamp, tenant, and user IDs; data containing "Accessing secrets".
Scope check failed	Message containing a time stamp, tenant, and user IDs; data containing "Authorization failed for `<scope>`".
Scope check failed	Message containing a time stamp, tenant, and user IDs; data containing "Authorization failed for `<scope>`".
Successful onboarding	Message containing a time stamp, tenant, and user IDs; data containing "Successful onboarding of tenant: `<tenant subdomain>`".
Failed Onboarding	Message containing a time stamp, tenant, and user IDs; data containing "Failed onboarding of tenant: `<tenant subdomain>`".
Successful Offboarding	Message containing a time stamp, tenant, and user IDs; data containing "Successful offboarding of tenant: `<tenant subdomain>`".
Failed Offboarding	Message containing a time stamp, tenant, and user IDs; data containing "Failed offboarding of tenant: `<tenant subdomain>`".
Scope check failed	Message containing a time stamp, tenant, and user IDs; data containing "Authorization failed for <scope>".
Read Connections Successful	Message containing a time stamp, tenant, and user IDs; data containing "Read for all connections successful".
Read Connections Failed	Message containing a time stamp, tenant, and user IDs; data containing "Failed to read for all connections".
Create Connection Successful	Message containing a time stamp, tenant, and user IDs; data containing "Created connection <name>".
Create Connection Successful	Message containing a time stamp, tenant, and user IDs; data containing "Failed to create connection <name>".
Delete Connection Successful	Message containing a time stamp, tenant, and user IDs; data containing "Deleted connection `<name>`".
Delete Connection Failed	Message containing a time stamp, tenant, and user IDs; data containing "Failed to delete connection `<name>`".

What Events Are Logged	How to Identify Related Log Events
Modify Connection	Message containing a time stamp, tenant, and user IDs; attributes containing the updated connection properties.
Failed read resource group	Data containing "Failed to read resource groups" or "Failed to read access Token for retrieving the resource groups"
Failed Create Disclaimer message	Data containing "Failed to create the disclaimer Message at AI Core tenant"
Failed Get Disclaimer Message	Data containing "Failed to retrieve the disclaimer message stored at AI Core tenant"
Failed Update Disclaimer Message	Data containing "Failed to update the disclaimer Message stored at AI Core tenant"
Failed Delete Disclaimer Message	Data containing "Failed to delete the disclaimer Message stored at AI Core tenant"
Delete Disclaimer Message	Message containing a time stamp, tenant id. Data containing "Disclaimer message Deleted"
Delete Admin Prompts	Message containing a time stamp, tenant, resource group, user prompt IDs; data containing "Prompt Schema Deleted"
Delete Prompt	Message containing a time stamp, tenant, resource group and prompt IDs; data containing "Prompt Data Deleted"
Delete Prompt Versions	Message containing a time stamp, tenant, resource group and prompt IDs, will also contain prompt versions if exists; data containing "Prompt Data Deleted"
Delete Prompt's Version	Message containing a time stamp, tenant, resource group and prompt IDs and prompt version data containing "Prompt Data Deleted"
Accessing Secrets	Data containing "Accessing secrets"
Deleting AI API Tenant Data	Message containing a time stamp, tenant, sub account and user IDs; data containing "Ai Api Tenant Data Deleted"
Clean Resource Groups	Message containing a time stamp, tenant, sub account and user IDs; data containing "All collections deleted for Resource Group Hash `hashID`"
Delete Workspace	Message containing a time stamp, tenant, sub account and user IDs; data containing "Workspace Secret Deleted: `name`"
Read Workspaces	Data containing "Read for all workspaces successful"
Failed Read Workspaces	Data containing "Failed to read for all workspaces"
Create Workspace for free plan	Data containing "Tenant having free plan is trying to create more than one connection"
Successful Create Workspace	Data containing "Created workspace `workspaceName`"
Soft Delete all Workspaces	Data containing "Soft deleted all workspaces"
Failed Soft Delete all Workspaces	Data containing "Failed to soft delete all workspaces"

What Events Are Logged	How to Identify Related Log Events
Failed Read specific Workspace	Data containing "Failed to read workspace `workspaceName`"
Delete specific workspace	Data containing "deleted workspace `workspaceName`"
Failed Delete specific workspace	Data containing "Failed to delete workspace `workspaceName`"
Failed to read meta api	Data containing "Failed to read meta api response"
Failed to fetch navigation items	Data containing "Failed to fetch navigation items"
Read specific workspace	Data containing "Read successful for workspace `workspaceName`"

The following information is described in the table columns:

- *Event grouping* - Events that are logged with a similar format or are related to the same entities.
- *What events are logged* - Description of the security or data protection and privacy related event that is logged.
- *How to identify related log events* - Search criteria or key words, that are specific for a log event that is created along with the logged event.
- *Additional information* - Any related information that can be helpful.

Related Information

[Audit Logging in the Cloud Foundry Environment](#)
[Audit Logging in the Neo Environment](#)
Roles and Authorizations [page 321]

9 Accessibility Features in SAP AI Launchpad

To optimize your experience of SAP AI Launchpad, SAP AI Launchpad provides features and settings that help you use the software efficiently.

> ⓘ Note
>
> SAP AI Launchpad is based on SAPUI5. For this reason, accessibility features for SAPUI5 also apply. See the accessibility documentation for SAPUI5 on SAP Help Portal at Accessibility for End Users.

For more information on screen reader support and keyboard shortcuts, see Screen-Reader Support for SAPUI5 Controls and Keyboard Handling for SAPUI5 Elements.

10 Monitoring and Troubleshooting

Getting Support

If you encounter an issue with this service, we recommend that you follow the procedure below.

Check Platform Status

Check the availability of the platform at SAP Trust Center🔗.

For more information about platform availability, updates and notifications, see Platform Updates and Notifications in the Cloud Foundry Environment.

Check Guided Answers

In the SAP Support Portal, check the Guided Answers🔗 section for SAP Business Technology Platform. You can find solutions for general issues as well as for specific services there.

Contact SAP Support

You can report an incident or error through the SAP Support Portal🔗.

Please use the following component(s) for your incident:

Component Name	Component Description
CA-ML-AILP	AI Launchpad

We recommend that you include the following information when you submit the incident:

- Region information (for example, EU10 or US10)
- Subaccount technical name
- URL of the page where the incident or error occurs
- Steps or clicks used to replicate the error
- Screenshots, videos, or the code entered

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon 🔗 : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:

  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.

  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.

- Links with the icon 🔗 : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.