

User Guide

Amazon SageMaker Unified Studio



Amazon SageMaker Unified Studio: User Guide

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

.....	xiii
What is Amazon SageMaker Unified Studio?	1
How does it work?	1
Terminology and concepts	2
Getting started	15
Access Amazon SageMaker Unified Studio	15
Configure credentials	16
Create a project	17
Get started with Amazon Bedrock in SageMaker Unified Studio	19
Chat with a model in the chat playground	19
Get started with the query editor	22
Prerequisites	23
Query sample data using Amazon Athena in Amazon SageMaker Unified Studio	23
Use the sample notebook	25
Projects	27
Create a new project	28
Step 1: Project name and description	28
Step 2: Customize parameters	29
Step 3: Review	30
Next steps	30
Get project details	31
View the SageMaker AI domain details associated with your project	31
Edit a project	32
Update a project	33
Move a project	34
Delete a project	35
Add project members	36
Remove project members	37
Bringing existing resources into Amazon SageMaker Unified Studio	39
AWS Glue Data Catalogs	39
Amazon S3 data	39
Amazon Athena workgroups and saved queries	39
Amazon EMR on EC2 clusters	39
Amazon Redshift clusters and Amazon Redshift Serverless workgroups	40

Amazon SageMaker AI resources	40
AWS IAM roles	40
Bring your own image (BYOI)	41
Dockerfile specifications	41
Dockerfile example	42
How to BYOI	43
Prerequisites	43
Step 1: Create your custom image	44
Step 2: Get the SageMaker AI domain name associated with your Amazon SageMaker Unified Studio project	44
Step 3: Attach your custom image using the SageMaker AI domain	44
Step 4: Access your custom image in Amazon SageMaker Unified Studio	44
Launch your custom image in Amazon SageMaker Unified Studio	44
Detach and clean up custom image resources	45
Using Amazon SageMaker Unified Studio Library for Python	46
Using ClientConfig	46
Domain	46
Domain Properties	47
Project	47
Project properties	47
S3 Path	48
MLflow Tracking Server ARN	49
Connections	49
Execution APIs	52
Local Execution APIs	53
Remote Execution APIs	55
IDE spaces	58
Creating a new space	58
Editing a space	59
Deleting a space	60
Using the coding assistant	60
JupyterLab	63
Managing configurations	65
Accessing metadata	68
Performing Git operations	69
Using the Amazon Q data integration in AWS Glue	69

Running SQL and Spark code	70
Visualizing results	71
Code Editor	71
Performing Git operations	72
Checking the version	73
Connections and extensions	73
Data	74
S3 data	74
Adding Amazon S3 data	75
Data catalog	81
Create a business glossary	82
Edit a business glossary	83
Delete a business glossary	84
Create a term in a glossary	85
Edit a term in a glossary	86
Delete a term in a glossary	87
Create a metadata form	88
Edit a metadata form	88
Delete a metadata form	89
Create a field in a metadata form	90
Edit a field in a metadata form	91
Delete a field in a metadata form	91
Data products	92
Create new data products	93
Publish data products	94
Edit data products	95
Unpublish data products	96
Delete data products	97
Subscribe to a data product	97
Review a subscription request and grant a subscription to a data product	98
Republish data products	99
Data inventory and publishing	100
Configure Lake Formation permissions for Amazon SageMaker Unified Studio	101
Create custom asset types	102
Create a data source for AWS Glue	104
Create a data source for Amazon Redshift	106

Edit a data source	108
Delete a data source	109
Publish assets to the catalog from the project inventory	109
Manage inventory and curate assets	110
Manually create an asset	112
Unpublish an asset from the catalog	113
Delete an asset	114
Manually start a data source run	114
Asset versioning	115
Data quality	116
Data lineage	119
Analyze your subscribed data with external analytics applications via JDBC connection	129
Metadata enforcement rules for publishing	132
Data discovery, subscription, and consumption	133
Search for and view assets in the catalog	134
Request subscription to assets	135
Approve or reject a subscription request	136
Revoke an existing subscription	138
Cancel a subscription request	138
Unsubscribe from an asset	139
Grant access to managed AWS Glue Data Catalog assets	140
Grant access to managed Amazon Redshift assets	141
Grant access for approved subscriptions to unmanaged assets	142
Metadata enforcement rules for subscription requests	142
Fine-grained access control to data	143
Limitations	144
Create row filters	145
Create column filters	146
Delete row or column filters	146
Edit row or column filters	147
Grant access with filters	148
Amazon SageMaker Lakehouse	150
How it works	151
Key components	152
Data connections	154
Capabilities	154

Supported data sources	154
Using Amazon SageMaker Lakehouse connections	155
Understanding created AWS resources	155
Getting started	156
Prerequisites	156
Create a project	157
Browse data	157
Upload data	158
Query data	158
Adding data sources	159
Create new connection	159
Upload data	162
Create a catalog	163
Add an existing databases and catalogs	163
Amazon S3 tables integration	164
Publishing data	166
Compute	168
Amazon Redshift	169
Gaining access to Amazon Redshift resources	169
Connecting to an existing Amazon Redshift resource	174
Creating a new Amazon Redshift Serverless compute resource	175
Removing an Amazon Redshift compute connection	176
Amazon EMR on EC2	176
Adding a new Amazon EMR on EC2 cluster	177
Adding an existing Amazon EMR on EC2 cluster	177
Using an Amazon EMR on EC2 cluster	184
Monitoring Amazon EMR on EC2 clusters	185
Terminating and removing an Amazon EMR on EC2 cluster	185
Spark History Server	186
EMR Serverless	186
Adding a new EMR Serverless application	186
Configuring permission mode	187
Deleting applications	188
Glue ETL	188
Configuring permission mode	189
Limitations of fine-grained permission mode	190

Visual ETL	192
Key features	192
Creating a Visual ETL flow	193
Supported connectors for Visual ETL	194
Supported transforms for Visual ETL	195
Add current timestamp transform	196
Aggregate transform	196
Array to columns transform	198
Autobalance processing transform	199
Change columns transform	199
Concatenate columns transform	200
Custom code transform	201
Derived column transform	202
Drop columns transform	203
Drop duplicates transform	204
Drop nulls transform	207
Explode array or map into rows transform	208
Extract JSON path transform	210
Fill nulls transform	210
Filter transform	211
Flatten transform	212
Format timestamp transform	213
Identifier transform	213
Intersect transform	214
Join transform	214
Limit tranform	215
Lookup transform	216
Parse JSON column transform	216
Pivot rows into columns transform	217
Regex extract transform	217
Rename columns transform	218
Select columns transform	219
Split string transform	220
SQL query transform	221
To timestamp transform	222
Union transform	222

Unpivot columns into rows transform	223
UUID transform	224
Authoring a visual ETL flow using generative AI	224
Scheduling and running visual flows	226
Scheduling visual flows from the editor	226
Reviewing scheduled visual flows in the editor	227
Scheduling visual flows with workflows	227
Using both external data and fine-grained data	231
Best practices for Visual ETL in Amazon SageMaker Unified Studio	233
Using Amazon Q with Amazon SageMaker Unified Studio	234
Machine learning	235
Discover Jumpstart models	237
Build models in JupyterLab	238
Train models	239
Use inference endpoints to deploy models	239
Create an endpoint and deploy a model	240
View your endpoints	241
Pipelines	241
Pipeline actions	242
Model registry	246
Create a model group	247
Create a collection	247
Register a model version	248
Track experiments using MLflow	248
MLflow Tracking Servers	249
Tracking experiments with MLflow	251
HyperPod clusters	251
Connect to a HyperPod cluster	252
View the HyperPod clusters	252
View details about a HyperPod cluster	253
HyperPod task governance	253
Open the HyperPod in JupyterLab	253
Partner AI apps	254
Amazon Bedrock in SageMaker Unified Studio	255
Discover Amazon Bedrock in SageMaker Unified Studio playgrounds	255
Build generative AI apps	256

Find serverless models with the model catalog	257
What is a prompt?	258
Inference parameters	259
System instructions	261
Prompt engineering guides	261
Experiment with the playgrounds	262
Chat with a model in the chat playground	263
Chat with an app in the chat playground	267
Generate an image with the image and video playground	268
Generate a video	277
Access shared generative AI assets in a playground	280
Build a chat agent app	281
Create a chat agent app with Amazon Bedrock	282
Deploy a chat agent app	291
Share a chat agent app	294
Build a flow app	296
Create a flow app with Amazon Bedrock	298
Define inputs with expressions	308
Use logic nodes to control flow	313
Use a chat agent app in a flow app	316
Flow nodes	318
Reuse and share prompts	325
Create a prompt	326
Add a prompt to a flow app	329
Modify a prompt	330
Delete a prompt	331
Share a prompt version	332
Evaluate the performance of a model	333
Create a model evaluation job	334
Model evaluation task types	337
Prompt datasets for model evaluation	345
Review a model evaluation job	349
Add a data source to your Amazon Bedrock app	352
Document data source	353
Web crawler data source	354
Content chunking and parsing	357

Create a Knowledge Base component	358
Add a Knowledge Base component to a chat agent app	360
Add a Knowledge Base component to a flow app	361
Synchronize a Knowledge Base	363
Safeguard your app with a guardrail	364
Guardrail policies	364
Create a guardrail component	367
Add a guardrail component to a chat agent app	369
Add a guardrail component to a flow app	369
Call functions from your Amazon Bedrock chat agent app	372
Function schema	372
Authentication methods	376
Create a function component	377
Add a function component to an Amazon Bedrock chat agent app	378
Use app history to view and restore app versions	379
Use your app outside of Amazon SageMaker Unified Studio	380
App export files	381
Export your app	383
Deploy an exported app	383
Run a deployed app	387
SQL analytics	389
Navigate the query editor	389
Connect data resources	390
Supported query engines	390
Create a query	390
Generative SQL	391
Review query history	393
Filtering the query history	393
Reviewing additional details	394
Schedule a query	394
Create a schedule	394
Review scheduled queries	395
Workflows	397
Create a workflow	397
Prerequisites	397
Create a workflow	398

Sample workflow	399
View workflow details	401
Run a workflow	402
Share a workflow	403
Workflow environments	404
Create a workflow environment	404
Update a workflow environment	405
Delete a workflow environment	405
Chat assistant	407
Security	409
Quotas	410
Troubleshooting	412
Troubleshooting AWS Lake Formation permissions for Amazon SageMaker Unified Studio	412
Amazon EBS Volume Depletion with Local Notebook Execution	414
Domain	415
SAML Identity Provider Email Issue	415
Project Creation Failure	415
Data Explorer Visibility Issue	416
Data Catalog Visibility Issue	416
Connection to Amazon RDS MySQL in Existing VPC	417
Visual ETL Flow Column Selection	417
Invalid or expired auth token when accessing an IDE	417
JupyterLab Configure Magic Error	418
Document history	419

What is Amazon SageMaker Unified Studio?

Amazon SageMaker Unified Studio is a unified development experience that brings together AWS data, analytics, artificial intelligence (AI), and machine learning (ML) services. It provides a place to build, deploy, execute, and monitor workflows from a single interface. This helps drive collaboration across teams and facilitate agile development.

In Amazon SageMaker Unified Studio, administrators manage the users and groups that can access Amazon SageMaker Unified Studio, and they set up resources for teams to use. When your administrator grants you access to Amazon SageMaker Unified Studio, you can contribute to Amazon SageMaker Unified Studio projects. Within Amazon SageMaker Unified Studio projects, you can collaborate on business use cases by creating and sharing data, computation work, and other resources.

This guide provides information about using Amazon SageMaker Unified Studio to create, contribute to, and manage projects. For more information about the administrator experience in the Amazon SageMaker Unified Studio management console, see the [Amazon SageMaker Unified Studio Administrator Guide](#).

How does it work?

After admins have set up and invited you to use the Amazon SageMaker Unified Studio portal, you can use the Amazon SageMaker Unified Studio portal to create and manage projects.

The flow for Amazon SageMaker Unified Studio users is as follows:

1. Gain access to the Amazon SageMaker Unified Studio portal by configuring your single sign-on (SSO) or IAM credentials and using the domain URL from your administrator.
2. Create a new project or navigate to a project that you have been added to.
3. Use the tools and resources within the Amazon SageMaker Unified Studio portal to build, share, and execute applications within your project.

Amazon SageMaker Unified Studio terminology and concepts

As you get started with Amazon SageMaker Unified Studio, it is important that you understand its key concepts, terminology, and components.

Amazon SageMaker Unified Studio

This is a browser-based web application where you can use all your data and tools for analytics and AI. Amazon SageMaker Unified Studio can authenticate you with your IAM user credentials or with credentials from your identity provider through the IAM Identity Center or with your SAML credentials. You can obtain the Amazon SageMaker Unified Studio URL for your domains by accessing the SageMaker AI management console at <https://console.aws.amazon.com/datazone>.

Amazon SageMaker AI management console

You can use the SageMaker AI management console at <https://console.aws.amazon.com/datazone> to access and configure your domains for user management, account associations, project profiles, blueprints, Amazon Bedrock models, Git connections, and Amazon Q usage.

Amazon Bedrock in SageMaker Unified Studio

Use Amazon Bedrock in SageMaker Unified Studio to build and scale generative AI applications. Amazon Bedrock in SageMaker Unified Studio provides a web interface that allow users to interact with [Amazon Bedrock](#) foundation models and use Amazon Bedrock tools, such as agents, guardrails, prompts, flows, evaluation, and functions in a seamless unified fashion. Users can interact with models in a generative AI playground or collaborate on developing generative AI applications in projects.

Amazon Q Developer

Amazon Q Developer is an AI coding assistant that can chat about code, provide inline code completions, generate net new code, scan your code for security vulnerabilities, and make code upgrades and improvements.

In the current release of Amazon SageMaker Unified Studio, by default, all users of an Amazon SageMaker Unified Studio domain have access to the Free Tier release of Amazon Q.

Amazon SageMaker Lakehouse

Amazon SageMaker Lakehouse unifies your data across Amazon S3 data lakes and Amazon Redshift data warehouses. Amazon SageMaker Lakehouse helps you build powerful analytics, machine learning (ML), and generative AI applications on a single copy of data.

Amazon SageMaker Lakehouse is accessible via Amazon SageMaker Unified Studio.

Amazon SageMaker Data Processing Visual ETL

In Amazon SageMaker Unified Studio you can author highly scalable extract, transform, load (ETL) data integration flows for distributed processing without becoming an Apache Spark expert. You can define your data integration flow in the simple visual interface and Amazon SageMaker Unified Studio automatically generates the code to move and transform your data. The code is generated in Python and written for Apache Spark. Additionally, you can choose to author your visual flows in English using generative AI prompts from Amazon Q.

Asset

In Amazon SageMaker Unified Studio, an asset is an entity that presents a single physical data object (for example, a table, a dashboard, a file) or virtual data object (for example, a view).

Asset type

Asset types define how assets are represented in the Amazon SageMaker catalog. An asset type defines the schema for a specific type of asset. When assets are created, they are validated against the schema defined by their asset type (by default, the latest version). When an asset update occurs, Amazon SageMaker Unified Studio creates a new asset version and enables Amazon SageMaker Unified Studio users to operate on all asset versions.

Associated accounts

Use account associations in Amazon SageMaker Unified Studio to publish data from other AWS accounts into the Amazon SageMaker catalog and create projects to work with data across multiple AWS accounts. AWS accounts where Amazon SageMaker unified root domains are created initiate the account association requests. You can request association from the Amazon SageMaker management console. Account association requests must be accepted by the administrators of the AWS accounts invited for account association. You can authorize the domain account to use data or allow infrastructure deployment with the right IAM permissions as part of approval. Once an associated account is linked to a domain, projects in Amazon SageMaker Unified Studio can use resources from those accounts and also other types of assets. You can deploy resources in specific AWS accounts through project profiles.

Authorization policy

Authorization policies are a set of controls within Amazon SageMaker Unified Studio applied to entities such as projects, blueprints, environments, glossary, and metadata forms.

Within an Amazon SageMaker Unified Studio domain unit, you can assign the following authorization policies to your users and groups to grant them specific permissions:

- Domain unit creation policy
- Project creation policy
- Project membership policy
- Domain unit ownership assumption policy
- Project ownership assumption policy

Within an Amazon SageMaker Unified Studio domain unit, you can assign the following authorization policies to your projects to grant them specific permissions:

- Glossary creation policy
- Metadata forms creation policy
- Custom asset type creation policy

Within a specific blueprint configuration, you can assign the following authorization policies to projects and domain unit owners:

- Create environment profiles using this blueprint. This policy can be assigned to Amazon SageMaker Unified Studio projects, and it authorizes them to create environment profiles using this blueprint.
- Grant permissions to create environment profiles using this blueprint . This policy can be assigned to domain unit owners and it authorizes them to grant permissions to projects to create environment profiles using this blueprint.

AWS account owner

In Amazon SageMaker Unified Studio, AWS account owners create roles, policies, and permissions in their AWS accounts that enable these AWS accounts to be associated with Amazon SageMaker Unified Studio domains.

Blueprint

A blueprint is used to create the project profile that defines which AWS tools and services project members can use as they work with data in the Amazon SageMaker catalog.

In the current release of Amazon SageMaker Unified Studio the following default blueprints are supported:

Blueprint name	Description	Resources created
AmazonBedrockGenerativeAI	This is the combined Amazon Bedrock blueprint which contains seven sub-Amazon Bedrock blueprints. Users create project profiles with this blueprint to build generative AI applications using tools such as agents, knowledge bases, guardrails, flows, functions, and model evaluation.	
AmazonBedrockChatAgent	Provides a reusable AWS CloudFormation template to create an Amazon Bedrock Agent and supporting resources, including an execution role and a consumption role.	Bedrock Agent, Bedrock Agent Execution role, Bedrock Agent Consumption role
AmazonBedrockEvaluation	Creates one IAM role as the service role for an Amazon Bedrock evaluation job.	Bedrock Evaluation job execution role
AmazonBedrockFlow	Provides a reusable AWS CloudFormation template to create an Amazon Bedrock Prompt Flow and supporting resources such as an execution role.	Amazon Bedrock Flow, Amazon Bedrock Flow Execution role
AmazonBedrockFunction	Provides a reusable AWS CloudFormation template	Secrets Manager secret, AWS Lambda function, AWS

Blueprint name	Description	Resources created
	to create an AWS Lambda function and supporting resources, such as an execution role, and a secret manager.	Lambda function execution role, Log group
AmazonBedrockGuardrail	Provides an AWS CloudFormation template to create an Amazon Bedrock Guardrail and supporting resources such as an execution role.	Amazon Bedrock Guardrail
AmazonBedrockKnowledgeBase	Provides an AWS CloudFormation template to create a reusable Amazon Bedrock Knowledge Base and supporting resources such as an execution role.	Amazon Bedrock Knowledge Base, OpenSearch Serverless collection, Amazon Bedrock Knowledge Base Execution role, AWS Lambdas, including OpenSearch Index Lambda and KB Ingestion Trigger Lambda, AWS Lambda Execution role, Amazon Bedrock Knowledge Base data source
AmazonBedrockPrompt	Provides a reusable AWS CloudFormation template to create an Amazon Bedrock Prompt and supporting resources, such as an execution role, and a consumption role.	Amazon Bedrock Prompt, Amazon Bedrock Prompt Consumption role

Blueprint name	Description	Resources created
DataLake	Provides a reusable AWS CloudFormation template to create a data lake environment with a AWS Glue database for data management and an Amazon Athena workgroup for querying data.	AWS Glue databases, lake formation permissions, Amazon Athena workgroups
EMRonEC2	Provides a reusable AWS CloudFormation template to create an Amazon EMR on EC2 cluster to run and scale Apache Spark, Hive, and other big data workloads.	EMR on EC2 clusters
EMRServerless	Provides a reusable AWS CloudFormation template to create an Amazon EMR Serverless application that is ready to serve Apache Spark batch jobs and interactive sessions.	EMR on Serverless applications
LakehouseCatalog	Provisions a new catalog in the Amazon SageMaker Lakehouse that is backed by Amazon Redshift Managed Storage	
MLExperiments	Provides OnDemand blueprint to enable MLflow tracking server for the experimentation inside a project.	MLflow tracking server (on demand)

Blueprint name	Description	Resources created
PartnerApps	Creates an IAM role and a Connection that enables access to Partner AI Apps. Through Partner AI Apps you can leverage integrated and fully-managed third-party solutions for AI/ML development.	Amazon SageMaker Partner AI Apps IAM role, Amazon SageMaker Partner AI Apps Connection
RedshiftServerless	Provides a reusable AWS CloudFormation template to create an Amazon Redshift Serverless environment to get insights from data without managing infrastructure.	Amazon Redshift Serverless warehouses
Tooling	Creates resources for the project, including IAM user roles, security groups, and Amazon SageMaker unified domains.	IAM user roles, Amazon SageMaker unified domains, security groups
Workflows	Provides an AWS CloudFormation template to create the MWAA environment for Airflow based Workflows	Enables project workflows on MWAA

Amazon SageMaker catalog

This is a catalog of all the published assets from various projects. The scope of the Amazon SageMaker catalog is the domain, therefore published assets are discoverable by all projects in that domain. The Amazon SageMaker catalog enables discovery that crosses the account and Region boundary. You can publish assets to the Amazon SageMaker catalog so that other projects can subscribe to them, or you can subscribe to assets in the catalog that were

published from other projects. Every asset that lives in the Amazon SageMaker catalog has an owner project (also known as the producer project) which controls policies around how subscriptions can be fulfilled. A subscriber (also known as a consumer project) can make a request to the owner project to gain access to the asset. Once the request is approved, the owner project provides the necessary permissions to the subscriber project so that it may gain access to that asset.

Business glossary

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms that may be associated with assets. A business glossary helps ensure that the same terms and definitions are used across an organization throughout its various data analytics tasks. You can add terms in a business glossary to assets and columns to classify or enhance the identification of those attributes during search. You can select glossary as the value type for a field in a metadata form that is associated with an asset. When you select a particular term as the value for an asset's metadata form field, users can search for the business glossary term and find the associated assets.

Git connection

Use git connections to check in and check out files and manage your code repository. When you create an Amazon SageMaker unified domain, a default git connection to CodeCommit is provided for you to manage your code. You can also create and enable new 3P Git connections to GitHub, GitHub Enterprise Server, GitLab, and GitLab Self-Managed.

Data source

An entity which brings in metadata from a source and adds metadata forms (such as the ingestion job). This entity allows publishers to capture ingestion configuration including what metadata forms to attach and whether to run BNG. Since this configuration has a one-to-many mapping with the credentials provided by the publisher, we believe that it should be captured in a separate entity.

In Amazon SageMaker Unified Studio, you can use data sources to import technical metadata of assets (data) from the source databases or data warehouses into Amazon SageMaker Unified Studio. In the current release of Amazon SageMaker Unified Studio, you can create and run data sources for AWS Glue and Amazon Redshift. By creating a data source, you establish a connection between Amazon SageMaker Unified Studio and the source (AWS Glue Data Catalog or Amazon Redshift Warehouse), which you can then use to read technical metadata, including table names, columns names, and data types. By creating a data source, you also begin the initial data source run that creates new or updates existing assets in Amazon SageMaker Unified

Studio. While creating a data source or after the data source is successfully created, you also have the option to specify a schedule for your data source runs.

Data source run

In Amazon SageMaker Unified Studio, a data source run is a task that Amazon SageMaker Unified Studio performs in order to create assets in project inventories and also optionally to publish project inventory assets to the Amazon SageMaker catalog. Data source runs can be automated (started when a data source is initially created), scheduled, or manual. Use data selection criteria to fine-tune the existing and future data sets to be ingested into project inventories or the Amazon SageMaker catalog and the frequency of metadata updates to those inventory or catalog assets.

Domain

In Amazon SageMaker Unified Studio, a domain is the organizing entity for connecting together your assets, users, and their projects. With Amazon SageMaker unified domains, you have the flexibility to reflect the data and analytics needs of your organizational structure, whether it's creating a single Amazon SageMaker unified domain for your enterprise or multiple domains for different business units.

Domain administrator

The IAM principal ID that has the super administrative permissions to edit entities in the domain.

In Amazon SageMaker Unified Studio, an IAM principal who creates an Amazon SageMaker Unified Studio domain is the default domain administrator of that domain. Domain administrators in Amazon SageMaker Unified Studio perform key functionalities for the domain, including creating domains, assigning other domain administrators, creating and managing project profiles, configuring blueprints, user management, account associations, Amazon Bedrock models, Git connections, and Amazon Q.

Domain unit

Use domain units to organize your assets and other domain entities under specific business units and teams. To set up secure and efficient data sharing within and across business units of your organization, you can create domain units within Amazon SageMaker Unified Studio and grant access to selected users within each business unit to log in and share their assets to the Amazon SageMaker catalog. Domain units can also be used for resource owners, such as AWS account owners, to set up Amazon SageMaker Unified Studio authorization permissions on their

resources. Domain units provide a delegated authority from account owners to domain unit owners and they can set up authorization permissions on behalf of account owners.

JupyterLab

Amazon SageMaker Unified Studio provides a JupyterLab interactive development environment (IDE) for you to use as you perform data integration, analytics, or machine learning in your projects. Amazon SageMaker Unified Studio notebooks are built on JupyterLab spaces and Amazon SageMaker Distribution.

Metadata form type

A metadata form type is a template that defines the metadata that is collected and saved when assets are created as inventory or published in an Amazon SageMaker unified domain. Metadata form types can be associated with a data asset. Metadata form types help domain administrators to define metadata forms needed for that domain, such as compliance information, regulation information, or classifications. Domain administrators can use this to customize additional metadata for their assets. Amazon SageMaker Unified Studio has system metadata form types such as asset-common-details-form-type, column-business-metadata-form-type, glue-table-form-type, glue-view-form-type, redshift-table-form-type, redshift-view-form-type, s3-object-collection-form-type, subscription-terms-form-type, and suggestion-form-type.

Metadata form

In Amazon SageMaker Unified Studio, metadata forms define the metadata that is collected and saved when assets are created as inventory or published in an Amazon SageMaker unified domain. Metadata form definitions are created in the domain catalog by a domain administrator. A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. A domain administrator applies a metadata form to assets in their domain by adding the metadata form to their domain. Asset publishers then provide any optional and required field values in the metadata form.

Project profile

In Amazon SageMaker Unified Studio, a project profile is a template for projects in your Amazon SageMaker unified domains. A project profile is a collection of blueprints, which are configurations used to create projects. A project profile can define if a particular blueprint is enabled during the creation of the project, or available later for the project users to enable on demand.

You must be an administrator of a Amazon SageMaker Unified Studio domain to create and manage project profiles. In the current release of Amazon SageMaker Unified Studio, you can create the following project profiles:

- Data analytics and AI/ML model development project profile
- SQL analytics project profile
- Generative AI application development project profile
- Custom project profile

Project

The project entity is the mechanism by which Amazon SageMaker Unified Studio users organize their work and provide business context over the jobs they are performing. A project is a container for all the user's code, including notebooks, queries, dashboards, workflows, and more. A project provides three capabilities:

1. Business context for the user's work which provides a level of audit to the functionality being performed.
2. A collaboration boundary where the users can work with each other by interacting with the project's source control repository.
3. A permissions boundary which gives users access to all the project artifacts and data/compute permissions after the users are added to the project.

A project exists within a domain. A single Amazon SageMaker unified domain can have several projects and each user can be added to multiple projects.

Each project is created using a template called a project profile, which is enabled by an administrator during the setup phase. A project profile controls the tools available within the project. Project members can request access to assets from the Amazon SageMaker catalog and produce new artifacts using one or more of the tools available inside the project. Artifacts in a project are not accessible outside of the project unless they are published to the Amazon SageMaker catalog which is discussed later.

Each project has one or multiple owners, who can add or remove other users (called project members) as owners or contributors and can modify or delete projects. Other restrictions on contributors can be defined with policies. When a user creates a project, they become the first owner of that project.

Project S3 path

The purpose of the project S3 path in Amazon SageMaker Unified Studio is to provide a secure, project-isolated location for storing temporary execution data and other project-related artifacts. The project S3 path follows a standardized structure of "<bucket>/<domain_id>/<project_id>/<project_scope>/" to ensure separation between projects and prevent objects from being shared across projects. The project S3 path is also used to store specific types of data, such as the location for the provisioned consumer AWS Glue database, Athena Workgroup output, and temporary storage for individual workflow runs.

Project Git repository

A project includes a dedicated git repository which serves as a central hub for users to manage version control for the code associated with their Amazon SageMaker Unified Studio projects. This enables collaboration across users within a project. All tools that generate file-based assets must use the project git repository for version control, for example, the query editor, JupyterLab IDE, and more. By default, Amazon SageMaker Unified Studio uses AWS CodeCommit as the project's repository which is created when a project is created. However, administrators can modify this to connect a third-party Git repository such as Github, Github Enterprise Server, GitLab, or BitBucket instead of the default repository.

Project member

A project member is any user who has been added to a project and given access to the project data and resources. Users can be enterprise users sourced from the IDP, or IAM Principals from one of the domain associated accounts. Project owners can add members either by adding them directly or by selecting enterprise groups. A project member is added to a project with a designation that defines the set of permissions it has within the project. Users can collaborate on various activities such as accessing data assets, performing data analysis or machine learning activities.

Subscription request

A request to use a data product.

In Amazon SageMaker Unified Studio, a subscription request is a process that an Amazon SageMaker Unified Studio project must follow in order to be granted access to a specific asset. Subscription requests can be approved, rejected, revoked, or granted.

Subscription grant

An object representing a fulfilled request for a particular project.

Querybook

Use querybooks to develop, run, and share multiple SQL queries in a single interactive notebook. They provide an environment for data scientists, analysts, and developers to query, analyze, and visualize data using Amazon Redshift or Amazon Athena as the query engine. Cells in a Querybook contain SQL statements or markdown and can be run individually, like a traditional query editor, or sequentially. Query results appear in-line with each cell, where you can toggle between multiple results and create data visualizations. To accelerate query development, Querybooks integrate with Amazon Q to generate SQL queries from natural language input, and provide auto-complete suggestions for table names, column names, and SQL keywords as you type. Amazon SageMaker Unified Studio automatically saves your work as you progress. When ready, you can publish your Querybook to your project for collaboration with teammates.

Space

A space in Amazon SageMaker Unified Studio refers to a personalized workspace that provides an isolated, sandboxed environment for users to run arbitrary code without interfering with other workers in a project. Each space consists of a compute instance, an EBS volume, and the JupyterLab application. Users can access their spaces through various entry points in Amazon SageMaker Unified Studio, the developer tools section, or by selecting Notebook files. The project Git repository is cloned into the space when you create the space. SageMaker Distribution is the image that is used to provide all the libraries, extensions, and packages in the IDE application.

Getting started

The information in this section helps you get started using Amazon SageMaker Unified Studio. If you are new to Amazon SageMaker Unified Studio, start by becoming familiar with the concepts and terminology presented in [*Terminology and concepts*](#).

To get started with Amazon SageMaker Unified Studio as a user, start by gaining access to Amazon SageMaker Unified Studio and creating a project. You can then add members to the project and use the sample JupyterLab notebook to begin building with a variety of tools and resources.

Topics

- [Access Amazon SageMaker Unified Studio](#)
- [Create a project](#)
- [Get started with Amazon Bedrock in SageMaker Unified Studio](#)
- [Get started with the query editor in Amazon SageMaker Unified Studio](#)
- [Use the sample notebook](#)

Access Amazon SageMaker Unified Studio

For you to get started with Amazon SageMaker Unified Studio, your admin must create a domain in the Amazon SageMaker Unified Studio console and provide you with a URL. For more information, see the Amazon SageMaker Unified Studio Administrator Guide.

When you have the URL from your admin, you can sign in to Amazon SageMaker Unified Studio in one of the following ways:

- By using your AWS IAM credentials. For more information, see [the section called “Sign up for an AWS account”](#).
- If your admin has configured single sign-on (SSO) access, you can also sign in to Amazon SageMaker Unified Studio using SSO credentials that you configure with IAM Identity Center or through an identity provider. For more information, see [the section called “Configure SSO credentials with IAM Identity Center”](#).

Note

Amazon SageMaker Unified Studio supports the following browsers:

Browser	Version
Microsoft Edge	Latest 3 major versions
Google Chrome	Latest 3 major versions
Apple Safari	Latest 3 major versions

JupyterLab IDE requires third-party cookies to be allowed in your Amazon SageMaker Unified Studio domain. For more information, see [the section called “Invalid or expired auth token when accessing an IDE”](#).

Configure credentials

If you want to sign in to Amazon SageMaker Unified Studio using AWS IAM user or SSO credentials using IAM Identity Center, follow the instructions in the optional prerequisite sections below.

 **Note**

You only need one method to sign in to Amazon SageMaker Unified Studio. If you have already configured an AWS account or SSO credentials that work with the domain URL you received from your admin, you can skip the steps in this section.

Topics

- [Sign up for an AWS account](#)
- [Configure SSO credentials with IAM Identity Center](#)

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

1. Open <https://portal.aws.amazon.com/billing/signup>.
2. Follow the online instructions.

When you sign up for an AWS account, an AWS account root user is created. The root user has access to all AWS services and resources in the account. As a security best practice, [assign administrative access to an administrative user](#), and use only the root user to perform [tasks that require root user access](#).

Configure SSO credentials with IAM Identity Center

You can use SSO with IAM Identity Center or with an identity provider. To use SSO with IAM Identity Center, work with your admin to get added to their IAM Identity Center directory and set up your SSO credentials.

The process is as follows:

1. After your admin adds your user information to their IAM Identity Center directory, you receive an email with your username and configuration instructions for single sign-on (SSO). Use the link in the email to set your password for SSO.
2. Your admin creates configurations and adds you to a domain using the Amazon SageMaker Unified Studio console. They then copy the link to that domain from the Amazon SageMaker Unified Studio console and send it to you. Use the domain URL from your admin to navigate to Amazon SageMaker Unified Studio.
3. Sign in to Amazon SageMaker Unified Studio with the SSO username and password that you configured in step 1.
4. If your admin's IAM Identity Center is configured to require multi-factor authentication (MFA), set up and use an MFA device. Follow the instructions on the screen to register or use an MFA device as needed, or contact your admin for support. For more information about MFA device enforcement, see [Configure MFA device enforcement](#) in the IAM Identity Center User Guide.

You are then able to view Amazon SageMaker Unified Studio landing page, where you can create new projects and view projects that you have been added to.

Create a project

In Amazon SageMaker Unified Studio, projects enable a group of users to collaborate on various business use cases. Within projects, you can manage data assets in the Amazon SageMaker Unified Studio catalog, perform data analysis, organize workflows, develop machine learning models, build generative AI apps, and more.

In order to create a project in Amazon SageMaker Unified Studio, you must gain access to Amazon SageMaker Unified Studio. A domain unit owner must also grant you access to create projects through an authorization policy. For more information, see [Domain units and authorization policies in Amazon SageMaker Unified Studio](#).

1. Navigate to the Amazon SageMaker Unified Studio landing page using the URL from your admin.

 **Note**

Amazon SageMaker Unified Studio supports the following browsers:

Browser	Version
Microsoft Edge	Latest 3 major versions
Google Chrome	Latest 3 major versions
Apple Safari	Latest 3 major versions

2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. Choose **Create project**.
4. Enter a name for your project. The name of the project is final.
5. (Optional) Enter a description for your project. You can edit this later.
6. (Optional) If your domain has configured domain units, select a domain unit for your project. If nobody in the domain has created domain units, you create a project in the root domain unit by default and no action is needed here.
7. Select the project profile that contains the resources you will need in your project.
 - a. Select **All capabilities** to access all of the supported services and resources in a single project.
 - b. Select **SQL analytics** to get started querying and analyzing SQL data.
 - c. Select **Generative AI application development** to get started with generative AI.
8. Choose **Continue**.

9. (Optional) Customize parameters, if desired. For more information about customizing parameters, see [the section called “Step 2: Customize parameters”](#).
10. Choose **Continue**.
11. Choose **Create project**.

You can then navigate to your project at any time from the Amazon SageMaker Unified Studio home page by choosing **Select a project** and **Browse all projects**, then choosing the name of your project. After you navigate to your project, you can begin adding data and compute resources and using tools.

Get started with Amazon Bedrock in SageMaker Unified Studio

Get started with Amazon Bedrock in SageMaker Unified Studio by experimenting with a model in a [playground](#).

The Amazon Bedrock in SageMaker Unified Studio playgrounds that lets you easily experiment with Amazon Bedrock models. The [chat](#) playground lets you chat with a model by providing text and image prompts to the model (not all models support images). The [image and video](#) playground lets you generate images and videos with a suitable model. With both playgrounds you can experiment by making configuration changes. For example, you can influence the response from a model by changing [inference](#) parameters.

After trying the chat and image playgrounds, you can try creating a chat agent app or flows app. A chat agent app allows users to chat with an Amazon Bedrock model through a conversational interface, typically by sending prompts (text or image) and receiving responses. To create a chat agent app, see [Build a chat agent app with Amazon Bedrock](#).

You can also create a [flows app](#) that lets you visually design the flow of an app.

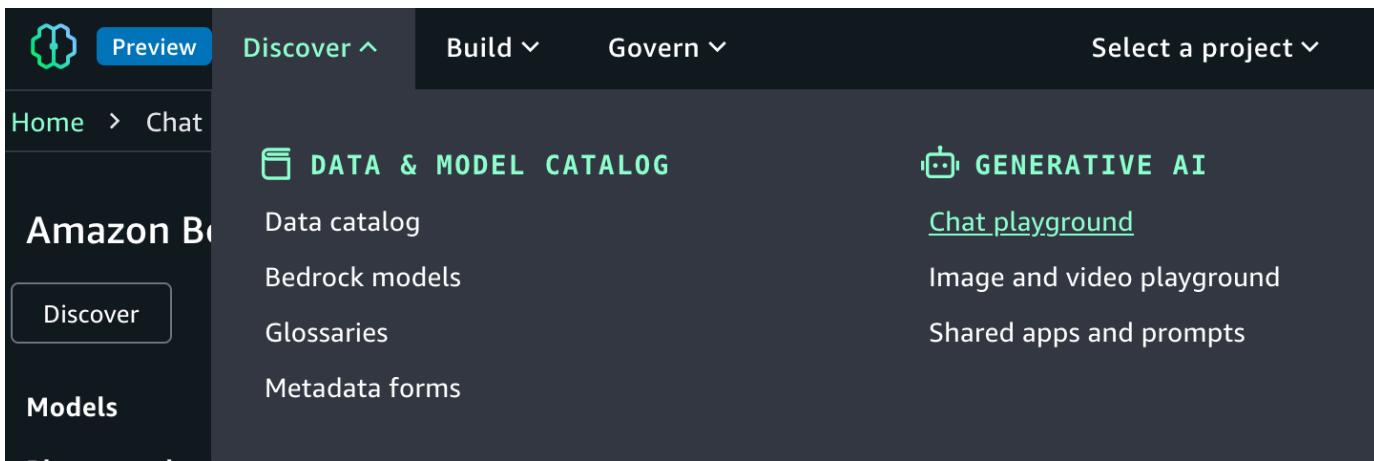
Chat with a model in the chat playground

In these instructions, you use the Amazon Bedrock in SageMaker Unified Studio chat playground to chat with an Amazon Bedrock in SageMaker Unified Studio model. You chat by sending a prompt to the model and answering the response that the model generates. For more information, see [Experiment with the Amazon Bedrock playgrounds](#).

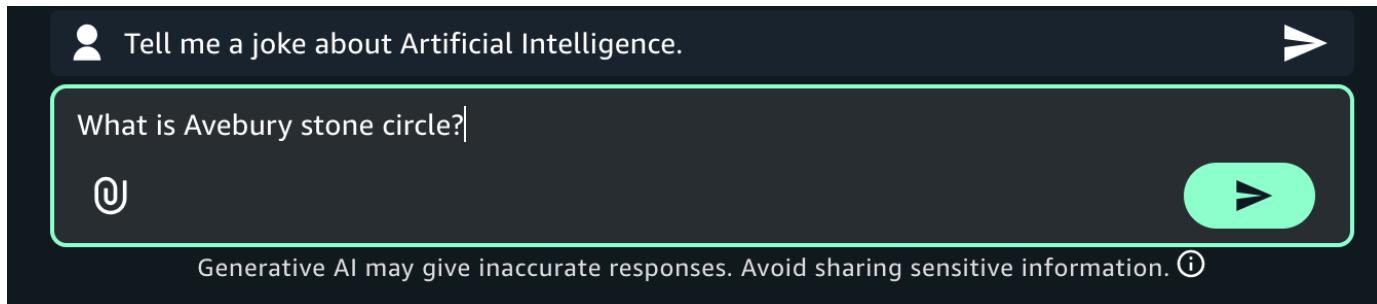
If you don't have access to a model, contact your administrator.

To chat with a model

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. At the top of the page, choose the **Discover**.
4. In the **Generative AI** section, choose **Chat playground** to open the chat playground.



5. In **Type** select **Model** and then select a model to use in **Model**. For full information about the model, choose **View full model details** in the information panel. For more information, see [Find serverless models with the Amazon Bedrock model catalog](#). If you don't have access to an appropriate model, contact your administrator. Different models might not support all features.
6. In the **Enter prompt** text box, enter **What is Avebury stone circle?**.
7. (Optional) If the model you chose is a reasoning model, you can choose **Reason** to have the model include its reasoning in the response. For more information, see [Enhance model responses with model reasoning](#) in the *Amazon Bedrock user guide*.
8. Press Enter on your keyboard, or choose the run button, to send the prompt to the model. Amazon Bedrock in SageMaker Unified Studio shows the response from the model in the playground.



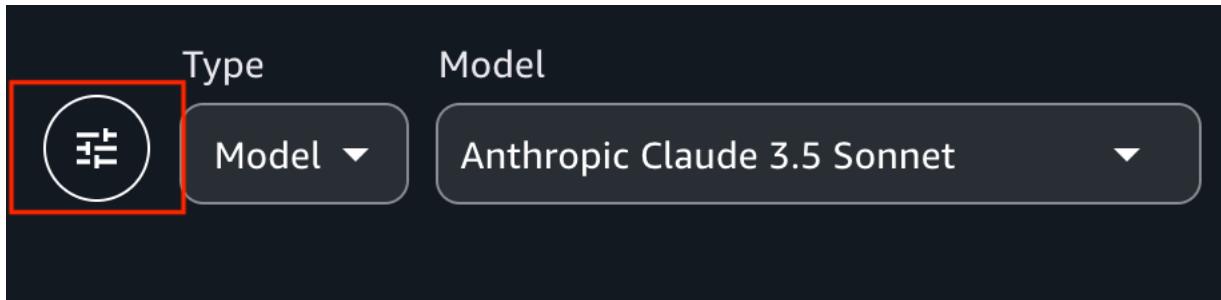
9. Continue the chat by entering the prompt **Is there a museum there?** and pressing Enter.

The response shows how the model uses the previous prompt as context for generating its next response.

10. Choose **Reset** to start a new chat with the model.

11. Influence the model response by doing the following:

- Enter and run a prompt. Note the response from the model.
- Choose the configurations menu to open the **Configurations** pane.



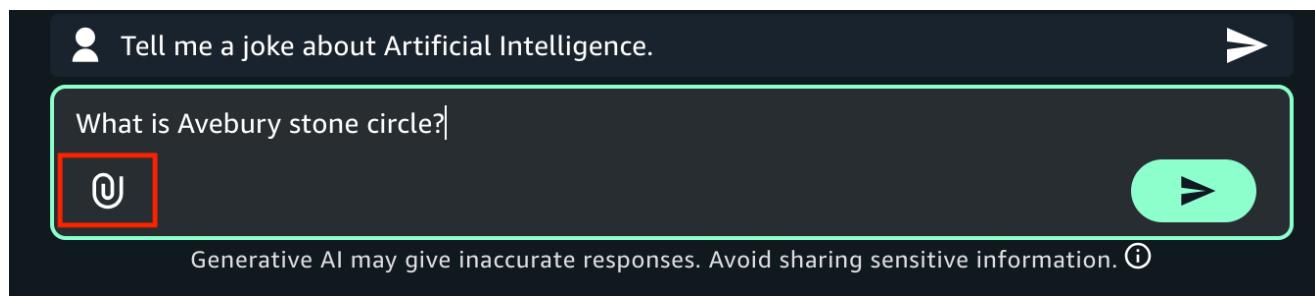
- Influence the model response by making [inference parameters](#) changes.
- (Optional) In **System instructions**, enter any overarching system instructions that you want the model to apply for future interactions.
- Run the prompt again and compare the response with the previous response.

12. Choose **Reset** to start a new chat with the model.

- 13.

14. Try sending an image to a model by doing the following:

- For **Model**, choose a model that supports [images](#).
- Choose the attachment button at the left of the **Enter prompt** text box.



- c. In the open file dialog box, choose an image from your local computer.
 - d. In the text box, next to the image that you uploaded, enter **What's in this image?**.
 - e. Press Enter on your keyboard enter to send the prompt to the model. The response from the models describes the model or image.
15. (Optional) Try using another model and different prompts. Different models have different recommendations for creating, or engineering, prompts. For more information, see [Prompt engineering guides](#).
16. (Optional) Compare the output from multiple models, or [shared apps](#).
- a. In the playground, turn on **Compare mode**.
 - b. In both panes, select the model that you want to compare. If you want to use a shared app, select **App** in **Type** and then select the app in **App**.
 - c. Enter a prompt in the text box and run the prompt. The output from each model is shown. You can choose the copy icon to copy the prompt or model response to the clipboard.
 - d. (Optional) Choose **View configs** to make configuration changes, such as [inference parameters](#). Choose **View chats** to return to the chat page.
 - e. (Optional) Choose **Add chat window** to add a third window. You can compare up to 3 models or apps.
 - f. Turn off **Compare mode** to stop comparing models.

Get started with the query editor in Amazon SageMaker Unified Studio

You can use the query editor to perform analysis using SQL. The query editor tool provides a place to write and run queries, view results, and share your work with your team.

Prerequisites

Before you get started with the query editor, you must access Amazon SageMaker Unified Studio and create a project with the **SQL analytics** project profile.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).

2. Create a project with a **SQL analytics** project profile. This project profile sets up your project with access to Amazon Redshift Serverless and Amazon Athena resources. For more information, see [the section called “Create a new project”](#).

Query sample data using Amazon Athena in Amazon SageMaker Unified Studio

After you create a project, you can use the query editor to write and run queries.

1. Navigate to the project you created in the top center menu of the Amazon SageMaker Unified Studio home page.
2. Expand the **Build** menu in the top navigation bar, then choose **Query editor**.
3. Create a new querybook tab. A querybook is a kind of SQL notebook where you can draw from multiple engines to design and visualize data analytics solutions.
4. Select a data source for your queries by using the menu in the upper-right corner of the querybook.
 - a. Under **Connections**, choose **Athena (Lakehouse)** to connect to your Lakehouse resources.
 - b. Under **Catalogs**, choose **AwsDataCatalog**.
 - c. Under **Databases**, choose the name of the AWS Glue database. This database was created for use when the project was created.
5. Choose **Choose** to connect to the database and query engine.
6. Copy the following SQL query into the querybook cell to create a table in the database.

```
CREATE TABLE mkt_sls_table AS
```

```
SELECT 146776932 AS ord_num, 23 AS sales_qty_sld, 23.4 AS wholesale_cost, 45.0 as  
lst_pr, 43.0 as sell_pr, 2.0 as disnt, 12 as ship_mode, 13 as warehouse_id, 23 as  
item_id, 34 as ctlg_page, 232 as ship_cust_id, 4556 as bill_cust_id  
UNION ALL SELECT 46776931, 24, 24.4, 46, 44, 1, 14, 15, 24, 35, 222, 4551  
UNION ALL SELECT 46777394, 42, 43.4, 60, 50, 10, 30, 20, 27, 43, 241, 4565  
UNION ALL SELECT 46777831, 33, 40.4, 51, 46, 15, 16, 26, 33, 40, 234, 4563  
UNION ALL SELECT 46779160, 29, 26.4, 50, 61, 8, 31, 15, 36, 40, 242, 4562  
UNION ALL SELECT 46778595, 43, 28.4, 49, 47, 7, 28, 22, 27, 43, 224, 4555  
UNION ALL SELECT 46779482, 34, 33.4, 64, 44, 10, 17, 27, 43, 52, 222, 4556  
UNION ALL SELECT 46779650, 39, 37.4, 51, 62, 13, 31, 25, 31, 52, 224, 4551  
UNION ALL SELECT 46780524, 33, 40.4, 60, 53, 18, 32, 31, 31, 39, 232, 4563  
UNION ALL SELECT 46780634, 39, 35.4, 46, 44, 16, 33, 19, 31, 52, 242, 4557  
UNION ALL SELECT 46781887, 24, 30.4, 54, 62, 13, 18, 29, 24, 52, 223, 4561
```

7. Choose the **Run cell** icon.



When the query finishes running, a **Result** tab appears below the cell to display the outcome.

8. Refresh the **Data explorer** navigation pane, and view the table you created in the **Lakehouse** section.
9. Choose **Add SQL** to add another cell to the querybook. Then enter the following script:

```
select * from mkt_sls_table limit 10
```

10. Choose the **Run cell** icon.

In the **Results** tab, the first ten rows of the table you created are displayed.

11. Choose **Add SQL** to add another cell to the querybook. Then enter the following script:

```
select item_id, sales_qty_sld  
from mkt_sls_table  
where sales_qty_sld > 30
```

12. Choose the **Run cell** icon.

In the **Results** tab, only data that fulfills the specified requirements is displayed.

13. In the **Results** tab, choose the **Chart view** icon.



This opens up a chart view with a line graph as a default.

14. Set up the chart to display a pie chart.

- a. For **Type**, choose **Pie**.
- b. For **Values**, choose **sales_qty_sold**.
- c. For **Labels**, choose **item_id**.

This displays a pie chart so you can visualize results.

After you've finished querying the data, you can choose to view the queries in your query history and save them to share with other project members.

- For more information about reviewing query history, see [the section called "Review query history"](#).
- For more information about other operations you can do with the query editor, such as using generative AI to create SQL queries, see [SQL analytics](#).

Use the sample notebook

You can get started using Amazon SageMaker Unified Studio by using the sample notebook in the JupyterLab IDE within your project. This `getting_started.ipynb` notebook provides information about using AWS Glue, Amazon Redshift, Amazon Athena, and more. This is a multi-service, poly-compute notebook, designed to enable end-to-end development in a single notebook.

In an Amazon SageMaker Unified Studio notebook, you can select the language and framework for each cell based on the compute options or connections configured in your project. You can add or modify these compute connections from the project's compute management screen. The compute choices differ based on your project's profile. However, all default profiles come with local Python, serverless Spark powered by AWS Glue, and Trino with Amazon Athena. There is a README file with additional information about the sample notebook and Amazon SageMaker Unified Studio.

You can also create new notebooks to input new code from scratch. For more information about using the JupyterLab IDE in Amazon SageMaker Unified Studio, see [the section called "JupyterLab"](#).

To navigate to the sample notebook, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to a project. To do this, choose **Select project** from the center menu.
3. Expand the **Build** menu, then choose **JupyterLab**.

Projects

A project in Amazon SageMaker Unified Studio is a boundary within a domain where you can collaborate with other users to work on a business use case. In projects, you can create and share data and resources. When you are added to a project, you gain access to relevant files and tools within that project, such as resources to create and deploy machine learning models. You can create new files and resources for the project, manage existing resources, and share work with other users on the same project.

After you gain access to Amazon SageMaker Unified Studio, you can create a new project or become added to an existing project by another user. The tools and resources available in the project are based on the project profile that is used to create a project. For more information about accessing Amazon SageMaker Unified Studio, see [the section called “Access Amazon SageMaker Unified Studio”](#).

There are two types of domains that your admin can create in the Amazon SageMaker Unified Studio management console. If your admin grants you access to an Amazon DataZone domain, see [Amazon DataZone projects and environments](#) in the Amazon DataZone User Guide. The sections in this guide describe projects in an Amazon SageMaker unified domain.

 **Note**

To join a project, you must create a project yourself or be added to a project by a project owner.

Topics

- [Create a new project](#)
- [Get project details](#)
- [Edit a project](#)
- [Update a project](#)
- [Move a project to a different domain unit](#)
- [Delete a project](#)
- [Add project members](#)

- [Remove project members](#)

Create a new project

In Amazon SageMaker Unified Studio, projects enable a group of users to collaborate on various business use cases. Within projects, you can manage data assets in the Amazon SageMaker Unified Studio catalog, perform data analysis, organize workflows, develop machine learning models, build generative AI apps, and more.

In order to create a project in Amazon SageMaker Unified Studio, you must gain access to Amazon SageMaker Unified Studio. A domain unit owner must also grant you access to create projects through an authorization policy. For more information, see [Domain units and authorization policies in Amazon SageMaker Unified Studio](#).

When you create a project, you choose a name and description, customize parameters for project resources, and then review selections.

Step 1: Project name and description

To begin creating a project, navigate to the Amazon SageMaker Unified Studio landing page and choose **Create project**.

The Project name and description includes the following fields:

- Project name - The name of your Amazon SageMaker Unified Studio project. Enter a name here. The name of the project can not be edited after the project is created.
- Description - An optional description of your project. You can edit this later.
- Domain unit - The business-level entity that lets your team organize and manage policies for business needs in the project. If nobody in the domain has created domain units, you create a project in the root domain unit by default and no action is needed here. If domain units have been created, select the name of the domain unit you want your project to be in. For more information, see [Domain units and authorization policies in Amazon SageMaker Unified Studio](#).
- Project profile - Project profiles define which resources and tools should be provisioned in the project. These include tools and compute resources for SQL, data science, data engineering, and machine learning development. Project profiles can include resources and tools from Amazon Redshift, Amazon SageMaker AI, and other AWS services. Select the project profile that contains the resources and tools you will need to use in your project. The project profiles available

for you to choose from are defined by your administrator in the Amazon SageMaker Unified Studio management console. For more information, see the [Amazon SageMaker Unified Studio Administrator Guide](#).

After you fill in the fields for project creation, choose **Continue** to customize parameters.

Step 2: Customize parameters

On the next page of project creation, you can view and edit the names and values for different resources that are created when the project is created.

Note

Some of the parameter values might be determined by your admin or by the default value from the environment blueprint, according to the configurations that your admin has set in the Amazon SageMaker Unified Studio management console. If you are not able to view or change a parameter value that you want to specify, contact your admin to edit the configurations. For more information, see the section [Edit a project profile](#) in the [Amazon SageMaker Unified Studio Administrator Guide](#).

Connect to a Git repository

As part of this process, if your admin has configured the parameters to be editable, you can choose a Git repository to connect to your project. You can choose to connect your project to an existing third-party Git repository or create a new Git repository to connect to.

To connect to an existing 3P Git repository

1. In the Git connection dropdown, select a connection from AWS CodeConnections that is enabled for Amazon SageMaker Unified Studio. Available Git connections are provided by your administrator in the Amazon SageMaker Unified Studio management console.
2. Select either the Existing repository and existing branch or Existing repository and new branch radio button. Then in the Repository name dropdown, you will see a list of repositories accessible with the connection.
3. Select the name of the repository you want to connect your project to.
4. In the branch name dropdown, either select an existing branch, or enter a branch name to create a new one.

To create a new Git repository

1. In the Git connection dropdown, select a connection from AWS CodeConnections that is enabled for Amazon SageMaker Unified Studio. Available Git connections are provided by your administrator in the Amazon SageMaker Unified Studio management console.
2. Select the New repository and new branch radio button.
3. Provide a name for the repository.
4. Enter a name for the branch you want to create within the new repository. The new repository and branch will then be created when the project is created.

Depending on which project profile you are using to create a project and what parameters your admin has configured to be editable, you might have other fields to choose parameters for.

When you have chosen the parameters you want, choose **Continue** to review the selections.

Step 3: Review

Use the last page of project creation to review the configurations you have selected. When everything is configured as desired on the project creation review page, choose **Create project**.

You are then redirected to the project home page. It might take a few minutes before the project is created and you can access tools.

Next steps

After you create a project, you can add members and resources to the project and begin using tools. There are many ways to get started building your project, including the following options:

- Add members to your project to collaborate together. For more information, see [the section called "Add project members"](#).
- Add data to your project. For more information, see [Data](#).
- Add compute resources to your project. For more information, see [Compute](#).
- Find, train, and deploy machine learning models. For more information, see [Machine learning](#).
- Use Amazon Bedrock in SageMaker Unified Studio to create generative AI apps. For more information, see [Amazon Bedrock in SageMaker Unified Studio](#).

Get project details

An Amazon SageMaker Unified Studio project contains essential configuration details that you may need to access for various development, deployment, and administrative tasks. The following section provides information on how to navigate to your **Project details** page, which includes key information and metadata. The **Project details** page includes:

- Project ID
- Project role ARN
- Amazon S3 location
- Domain unit name
- Domain ID

To navigate to your Project details page

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your IAM Identity Center (SSO) or AWS credentials. For more information, see [Access Amazon SageMaker Unified Studio](#).
2. If not open already, navigate to the Amazon SageMaker Unified Studio home page by choosing the icon located at the top left corner of the page.
3. Under **Your projects**, choose your project.
4. If not open already, choose the **Project details** tab.

View the SageMaker AI domain details associated with your project

A SageMaker AI domain is created for your Amazon SageMaker Unified Studio Project. You may need to update your SageMaker AI domain to update your Amazon SageMaker Unified Studio Project. Use the following instructions to get the associated SageMaker AI domain details.

 **Note**

The domain ID in [Get project details](#) is *not* the same as your SageMaker AI domain ID.

To get your SageMaker AI domain details

1. Navigate to your **Project details** tab. For more information, see [Get project details](#).
2. Copy your **Project ID**.
3. Navigate to the [Amazon Amazon SageMaker AI console](#).
4. Expand the **Admin configurations** section.
5. Under **Admin configurations**, choose **Domains**.
6. From the list of domains, find the domain name that contains the **Project ID** you copied above. You can use the search function on that page.
7. Choose the hyperlink of your domain to view your **Domain details**.

Edit a project

In Amazon SageMaker Unified Studio, projects enable a group of users to collaborate on various business use cases. Within projects, you can manage data assets in the Amazon SageMaker Unified Studio catalog, perform data analysis, organize workflows, develop machine learning models, build generative AI apps, and more.

To edit an Amazon SageMaker Unified Studio project, you must be the owner of that project or the domain administrator of the domain that contains this project. Project owners can edit the project in the following ways:

- Adding members to a project. For more information, see [the section called “Add project members”](#).
- Removing members from a project. For more information, see [the section called “Remove project members”](#).
- Updating project parameters. For more information, see [the section called “Update a project”](#).
- Changing a project description.

To change the project description in an existing project, complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select a project**.

3. If you don't see the name of your project under **Recently updated projects**, choose **Browse all projects**.
4. Choose the project that you want to edit. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Search projects** field.
5. On the **Project overview** page, choose the edit icon next to **Description**. An editable text box appears.
6. Input the changes you want to make in the text box.
7. Choose the checkmark to save your changes.

Update a project

You and your admin can make changes to project settings and parameters by updating a project. There are three different kinds of project updates:

- Service-initiated updates. Projects are made up of blueprints that set up resources using different AWS services, and sometimes these services require updates to their parameters due to feature changes or security initiatives.
- Admin-initiated updates. The Amazon SageMaker Unified Studio domain admin can edit or update blueprint parameters for a project in the Amazon SageMaker Unified Studio management console after that project has been created. When this happens, project owners in Amazon SageMaker Unified Studio must update the project for the changes to take effect.
- Owner-initiated updates. Amazon SageMaker Unified Studio project owners can update certain parameters. Some parameters are not modifiable, and other parameters are modifiable if the domain admins configure them to be editable in the Amazon SageMaker Unified Studio management console. The project owners can then make updates to modifiable parameters in Amazon SageMaker Unified Studio.

When a service-initiated update or admin-initiated update occurs and a project is eligible for updating, a blue banner appears to notify project owners of the change. Project owners can then update the project to implement the changes.

Note

If a service or admin has initiated an update by updating relevant blueprints, the project owner must update the project so that project members can use all project capabilities. The project capabilities might be limited until the project owner completes the update.

Project owners can initiate updates to some project parameters if their domain admin has made those parameters editable.

To update a project, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. From the top center menu, choose **Browse all projects**.
3. Select the name of the project to navigate to that project.
4. On the **Project overview** page, choose **Actions > Update project**.
5. (Optional) Review the information in the **Domain updates** section, if applicable.
6. Review the information in the **Project profile updates** section.
7. Review the information in **Environment updates**.
 - Choose **Show update parameters** in each section to review and update the modifiable parameters as desired. If there are other parameters that you want to update that are listed as non-modifiable, contact your admin.
8. Choose **Update project**. Updating a project might take a few minutes as resources are deployed.

When the update is complete, you can use your project with the updated parameters.

Move a project to a different domain unit

You can move a project from one domain unit to another. To move a project to a different domain unit in Amazon SageMaker Unified Studio, you must meet the following requirements:

- You must have a policy grant for project creation in the domain unit to which you are moving the project.

- All members of the project must have project membership permissions in the domain unit to which you are moving the project.
- You must be a domain unit owner in the domain unit to which you're moving the project.
- You must be an owner of the project.

To move an existing project to a different domain unit, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Browse projects**.
3. Choose the project that you want to move.
4. Expand **Actions** and choose **Move project**.
5. Select the domain unit that you want to move this project to and then choose **Move project**.

Delete a project

The act of deleting a project is final. Deletion irrevocably deletes the project's contents, including compute instances, data sources, queries, and more, and all the content within those project resources. Deleting a project does not delete non-Amazon SageMaker Unified Studio AWS resources that Amazon SageMaker Unified Studio might have helped you create. If you no longer need these AWS resources, delete them in their respective AWS service and account. For example, if you create Amazon Bedrock model evaluation jobs in Amazon Bedrock in SageMaker Unified Studio, the jobs that aren't automatically deleted when you delete the project. You will need to use the Amazon Bedrock console to delete the model evaluation jobs. Contact your administrator if you don't have access to the Amazon Bedrock in SageMaker Unified Studio console.

To delete an Amazon SageMaker Unified Studio project, you must be an owner of the project.

To delete an existing project, complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select a project**.
3. If you don't see the name of your project under **Recently updated projects**, choose **Browse all projects**.

4. Choose the project that you want to delete. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Search projects** field.
5. On the **Project overview** page, expand **Actions** and choose **Delete project**.
Review the informational warnings about the potential impact of deleting the project.
6. If you accept the warnings, then type in the confirmation text and choose **Delete project**.

 **Important**

Deleting a project is an irrevocable action that cannot be undone by you or by AWS.

Add project members

In Amazon SageMaker Unified Studio, projects enable a group of users to collaborate on various business use cases. Within projects, you can manage data assets in the Amazon SageMaker Unified Studio catalog, perform data analysis, organize workflows, develop machine learning models, build generative AI apps, and more.

A project member can be a user or a group of users, and a project can have a maximum of 20 members.

There are two different roles that members can have:

- **Contributor:** These members can contribute to the project.
- **Owner:** These members can contribute to the project, add members to the project, and remove members of the project. They can also edit the project description or delete the project. The person who creates the project has the role of Owner by default.

There are three different kinds of members you can add:

- Single sign-on (SSO) users. An SSO user can sign into Amazon SageMaker Unified Studio using credentials from IAM Identity Center or another SSO source.
- SSO groups. You can add groups of users created in IAM Identity Center. An SSO group is considered one project member.

- IAM principals (roles or users). An IAM user can sign into Amazon SageMaker Unified Studio with their IAM credentials. Note that IAM roles can't access Amazon SageMaker Unified Studio directly and must contribute to the project programmatically.

To add members to an existing project, complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select a project**.
3. If you don't see the name of your project under **Recently updated projects**, choose **Browse all projects**.
4. Choose the project that you want to edit. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Search projects** field.
5. On the **Project overview** page, expand **Actions** and choose **Manage members**.
6. Choose **Add members**.
7. Enter the name of the user or group you want to add in the search bar, and select the name from the list.
8. Select **Contributor** if you want to add the project member as a contributor, or choose **Owner** if you want to add the project member as a project owner.
9. (Optional) Repeat these steps to add more project members. You can add up to 8 project members at a time.
10. Choose **Add members**.

Remove project members

In Amazon SageMaker Unified Studio, projects enable a group of users to collaborate on various business use cases. Within projects, you can manage data assets in the Amazon SageMaker Unified Studio catalog, perform data analysis, organize workflows, develop machine learning models, build generative AI apps, and more.

A project member can be a user or a group of users, and a project can have a maximum of 20 members. To remove members from a project, you must be an owner of that project.

To remove members from an existing project, complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select a project**.
3. If you don't see the name of your project under **Recently updated projects**, choose **Browse all projects**.
4. Choose the project that you want to edit. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Search projects** field.
5. On the **Project overview** page, expand **Actions** and choose **Manage members**.
6. Select the member or members you want to remove from the project. You can remove up to 8 members at a time.
7. Choose **Remove members**. A popup window appears so that you can confirm this action.
8. Choose **Remove members**.

Members that you remove from the project will no longer have access to files and data in that project. The assets that were created by those project members remain in the project inventory.

Bringing existing resources into Amazon SageMaker Unified Studio

You can bring in existing resources to your Amazon SageMaker Unified Studio project by using the **Data** and **Compute** pages in your project, or by using scripts provided in GitHub.

Examples of resources you can bring into Amazon SageMaker Unified Studio are listed below.

AWS Glue Data Catalogs

- A GitHub script for bringing this resource into Amazon SageMaker Unified Studio can be found [here](#).

Amazon S3 data

To bring your existing Amazon S3 data and use it in Amazon SageMaker Unified Studio, follow the guide in the link below. This guide explains how you can configure permissions and customize role assignments for Amazon SageMaker Unified Studio to access your Amazon S3 data from a project.

- A GitHub script for bringing this resource into Amazon SageMaker Unified Studio can be found [here](#).

Amazon Athena workgroups and saved queries

- A GitHub script for bringing this resource into Amazon SageMaker Unified Studio can be found [here](#).

Amazon EMR on EC2 clusters

- A GitHub script for bringing this resource into Amazon SageMaker Unified Studio can be found [here](#).
- To bring in existing clusters using the Amazon SageMaker Unified Studio interface, see [the section called “Adding an existing Amazon EMR on EC2 cluster”](#).

Amazon Redshift clusters and Amazon Redshift Serverless workgroups

- To bring in existing Amazon Redshift clusters and workgroups using the Amazon SageMaker Unified Studio interface, see [the section called “Connecting to an existing Amazon Redshift resource”](#).

Amazon SageMaker AI resources

- Amazon SageMaker Studio users can bring their existing SageMaker AI domains, user profiles, and spaces into Amazon SageMaker Unified Studio. This integration also supports additional SageMaker AI resources such as training jobs, machine learning pipelines, models, inference endpoints, and more. A package of GitHub scripts for bringing Amazon SageMaker AI resources into Amazon SageMaker Unified Studio can be found [here](#).

AWS IAM roles

Use the utility script in GitHub to configure permissions and customize role assignments for Amazon SageMaker Unified Studio.

- A GitHub script for bringing this resource into Amazon SageMaker Unified Studio can be found [here](#).

Note

Review the prerequisites carefully before proceeding to execute the script. Ensure that you save your work and that you do not have any running tasks or processes such as reconfiguring a JupyterLab space or creating a new compute resource. These processes might get interrupted or cause the script to fail.

Bring your own image (BYOI)

An image is a file that identifies the kernels, language packages, and other dependencies required to run your applications. It includes:

- Programming languages (like Python or R)
- Kernels
- Libraries and packages
- Other necessary software

Amazon SageMaker AI Distribution ([sagemaker-distribution](#)) is a set of Docker images that include popular frameworks and packages for machine learning, data science, and visualization.

You can also create your own custom image, using an Amazon SageMaker AI Distribution image as a base image, to bring your own image (BYOI). You may want to BYOI when:

- You need a specific version of a programming language or library
- You want to include custom tools or packages
- You're working with specialized software not available in the standard images

Topics

- [Dockerfile specifications](#)
- [How to BYOI](#)
- [Launch your custom image in Amazon SageMaker Unified Studio](#)
- [Detach and clean up custom image resources](#)

Dockerfile specifications

To bring your own image (BYOI), ensure that the following specifications are satisfied when you create your Dockerfile.

- **Base image specifications:** You must use one of the base images from Amazon SageMaker AI Distribution ([sagemaker-distribution](#)), with the following specifications. These contain

important extensions that are required to execute an image on Amazon SageMaker Unified Studio.

- The base image must be FROM public.ecr.aws/sagemaker/sagemaker-distribution:*version*. You can copy the Image URI of an image from the Image tags tab in [sagemaker-distribution](#).
- The chosen image *version* must be greater or equal to the following.
 - For CPU: 2.6-cpu
 - For GPU: 2.6-gpu
- Follow the [Custom image specifications](#) in the *SageMaker AI Developer Guide*.

Dockerfile example

The following is an example Dockerfile that meets the above criteria. The *version* in this example must satisfy the specification above.

Note

Adding ENTRYPPOINT in the Dockerfile will not work as expected. If you would like to configure a custom entrypoint, you will need to update your [ContainerConfig](#). For an example, see [Update container configuration](#) in the *SageMaker AI Developer Guide*.

```
FROM public.ecr.aws/sagemaker/sagemaker-distribution:version

ARG NB_USER="sagemaker-user"
ARG NB_UID=1000
ARG NB_GID=100

ENV MAMBA_USER=$NB_USER

USER root

RUN apt-get update
RUN micromamba install sagemaker-inference --freeze-installed --yes --channel conda-forge --name base

USER $MAMBA_USER
```

How to BYOI

When you bring your own image (BYOI) to Amazon SageMaker Unified Studio, you attach a custom image to an Amazon SageMaker Unified Studio project. The following page provides instructions on how to bring your custom image to your Amazon SageMaker Unified Studio project.

Topics

- [Prerequisites](#)
- [Step 1: Create your custom image](#)
- [Step 2: Get the SageMaker AI domain name associated with your Amazon SageMaker Unified Studio project](#)
- [Step 3: Attach your custom image using the SageMaker AI domain](#)
- [Step 4: Access your custom image in Amazon SageMaker Unified Studio](#)

Prerequisites

You will need to complete the following prerequisites to bring your own image to Amazon SageMaker Unified Studio.

- Create an Amazon SageMaker Unified Studio project. For more information, see [Create a new project](#).
- Set up the Docker application. For more information, see [Get started](#) in the *Docker documentation*.
- Install the latest AWS CLI by following the steps in [Getting started with the AWS CLI](#) in the *AWS Command Line Interface User Guide for Version 2*.
- Permissions to access the Amazon Elastic Container Registry (Amazon ECR) service. For more information, see [Amazon ECR Managed Policies](#) in the *Amazon ECR User Guide*.
- An AWS Identity and Access Management role that has the [AmazonSageMakerFullAccess](#) policy attached.

Step 1: Create your custom image

Important

Ensure that you are using the [Dockerfile specifications](#) in the following instructions.

Follow the steps in [Create a custom image and push to Amazon ECR](#) in the *SageMaker AI Developer Guide*.

Step 2: Get the SageMaker AI domain name associated with your Amazon SageMaker Unified Studio project

An associated SageMaker AI domain is created when you create a Amazon SageMaker Unified Studio project. You will need the SageMaker AI domain name before proceeding to the next step. For instructions, see [View the SageMaker AI domain details associated with your project](#).

Step 3: Attach your custom image using the SageMaker AI domain

To attach your custom image to your Amazon SageMaker Unified Studio project, you must attach your custom image to your SageMaker AI domain. Follow the steps in [Attach your custom image to your domain](#) in the *SageMaker AI Developer Guide*, using the SageMaker AI domain obtained from above.

Step 4: Access your custom image in Amazon SageMaker Unified Studio

Once your custom image is attached to your Amazon SageMaker Unified Studio project, the users with access to your project can access it. For instructions on how users can access the custom images, see [Launch your custom image in Amazon SageMaker Unified Studio](#).

Launch your custom image in Amazon SageMaker Unified Studio

The following section provides information on how to launch your custom image in Amazon SageMaker Unified Studio. If you have not already created a custom image, see [How to BYOI](#).

To launch your custom image in Amazon SageMaker Unified Studio

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your IAM Identity Center (SSO) or AWS credentials. For more information, see [Access Amazon SageMaker Unified Studio](#).
2. Navigate to the Amazon SageMaker Unified Studio home page by choosing the icon located at the top left corner of the page.
3. You can choose to create a project or choose an existing one.
4. Once you choose a project you will be taken to the **Project overview** page.
5. On the **Project overview** page, open your notebook. To do so:
 - a. Choose **New** to expand the dropdown menu, located at the top right of the page.
 - b. Choose **Notebook**, then continue with the following steps.
6. Once your notebook is open, choose **Configure space**, located at the top right of the page.
7. In the **Configure space** section, you can choose the image from the **Image** dropdown menu.
8. After you have verified the settings, choose **Save and restart**.

Once completed, this will start a new JupyterLab instance with the custom image.

Detach and clean up custom image resources

An associated SageMaker AI domain is created when you create an Amazon SageMaker Unified Studio project. You will need the SageMaker AI domain details to detach and clean up your custom image resources. For instructions, see [View the SageMaker AI domain details associated with your project](#).

Follow the instructions in [Detach and clean up custom image resources](#) in the *Amazon SageMaker AI Developer Guide*.

Using Amazon SageMaker Unified Studio Library for Python

The Amazon SageMaker Unified Studio library is an open source library for interacting with Amazon SageMaker Unified Studio resources. With this library, you can access resources such as domains, projects, connections, and databases, all in one place with minimal code. The following examples demonstrate how to use the library in local and remote sessions.

Using ClientConfig

If using `ClientConfig` for supplying credentials or changing the AWS region name, the `ClientConfig` object will need to be supplied when initializing any further Amazon SageMaker Studio objects, such as `Domain` or `Project`. If using non-prod endpoint for an AWS service, it can also be supplied in the `ClientConfig`. Note: In sagemaker space, datazone endpoint is by default fetched from the metadata JSON file.

```
from sagemaker_studio import ClientConfig, Project
conf = ClientConfig(region="eu-west-1")
proj = Project(config=conf)
```

Domain

Domain can be initialized using the following command.

```
from sagemaker_studio import Domain
dom = Domain()
```

If you are not using the Amazon SageMaker Studio within Amazon SageMaker Unified Studio JupyterLab IDE, you will need to provide the ID of the domain you want to use.

```
dom = Domain(id="123456")
```

Domain Properties

A Domain object has several string properties that can provide information about the domain that you are using.

```
dom.id  
dom.root_domain_unit_id  
dom.name  
dom.domain_execution_role  
dom.status  
dom.portal_url
```

Project

Project can be initialized using the following command.

```
from sagemaker_studio import Project  
proj = Project()
```

If you are not using the Amazon SageMaker Studio library within the Amazon SageMaker Unified Studio JupyterLab IDE, you will need to provide either the ID or name of the project you would like to use and the domain ID of the project.

```
proj = Project(name="my_proj_name", domain_id="123456")
```

Project properties

A Project object has several string properties that can provide information about the project that you are using.

```
proj.id  
proj.name  
proj.domain_id,  
proj.project_status,  
proj.domain_unit_id,  
proj.project_profile_id  
proj.user_id
```

IAM Role ARN

To retrieve the project IAM role ARN, you can retrieve the `iam_role` field. This gets the IAM role ARN of the default IAM connection within your project.

```
proj.iam_role
```

AWS KMS Key ARN

If you are using a AWS KMS key within your project, you can retrieve the `kms_key_arn` field.

```
proj.kms_key_arn
```

S3 Path

One of the properties of a Project is `s3`. You can access various S3 paths that exist within your project.

```
# S3 path of project root directory  
proj.s3.root  
# S3 path of datalake consumer Glue DB directory (requires DataLake environment)  
proj.s3.datalake_consumer_glue_db  
# S3 path of Athena workgroup directory (requires DataLake environment)  
proj.s3.datalake_athena_workgroup
```

```
# S3 path of workflows output directory (requires Workflows environment)
proj.s3.workflow_output_directory
# S3 path of workflows temp storage directory (requires Workflows environment)
proj.s3.workflow_temp_storage
# S3 path of EMR EC2 log destination directory (requires EMR EC2 environment)
proj.s3.emr_ec2_log_destination
# S3 path of EMR EC2 log bootstrap directory (requires EMR EC2 environment)
proj.s3.emr_ec2_certificates
# S3 path of EMR EC2 log bootstrap directory (requires EMR EC2 environment)
proj.s3.emr_ec2_log_bootstrap
```

Other Environment S3 Paths

You can also access the S3 path of a different environment by providing an environment ID.

```
proj.s3.environment_path(environment_id="env_1234")
```

MLflow Tracking Server ARN

If you are using an MLflow tracking server within your project, you can retrieve the `mlflow_tracking_server_arn` field.

Usage

```
proj.mlflow_tracking_server_arn
```

Connections

You can retrieve a list of connections for a project, or you can retrieve a single connection by providing its name.

```
proj_connections: List[Connection] = proj.connections
proj_redshift_conn = proj.connection("my_redshift_connection_name")
```

Each Connection object has several properties that can provide information about the connection.

```
proj_redshift_conn.name  
proj_redshift_conn.id  
proj_redshift_conn.physical_endpoints[0].host  
proj_redshift_conn.iam_role
```

Connection data

To retrieve all properties of a Connection, you can access the data field to get a ConnectionData object. ConnectionData fields can be accessed using the dot notation (e.g. conn_data.top_level_field). For retrieving further nested data within ConnectionData, you can access it as a dictionary. For example: conn_data.top_level_field["nested_field"].

```
conn_data: ConnectionData = proj_redshift_conn.data  
red_temp_dir = conn_data.redshiftTempDir  
lineage_sync = conn_data.lineageSync  
lineage_job_id = lineage_sync["lineageJobId"]  
spark_conn = proj.connection("my_spark_glue_connection_name")  
id = spark_conn.id  
env_id = spark_conn.environment_id  
glue_conn = spark_conn.data.glue_connection_name  
workers = spark_conn.data.number_of_workers  
glue_version = spark_conn.data.glue_version
```

Secrets

Retrieve the secret (username, password, other connection-related metadata) for the connection using the following property.

```
snowflake_connection: Connection = proj.connection("project.snowflake")  
secret = snowflake_connection.secret
```

Secrets can be a dictionary containing credentials or a single string depending on the connection type.

Catalogs, databases, and tables

If your Connection is of the IAM type, you can retrieve catalogs, databases, and tables within a project.

Catalogs

If your Connection is of the IAM type, you can retrieve a list of catalogs or a single catalog by providing its id.

```
iam_conn: Connection
conn_catalogs: List[Catalog] = iam_conn.catalogs
my_catalog: Catalog = iam_conn.catalog("1234567890:catalog1/sub_catalog")
```

Each Catalog object has several properties that can provide information about the catalog.

```
my_catalog.name
my_catalog.id
my_catalog.type
my_catalog.spark_catalog_name
my_catalog.resource_arn
```

Databases

You can retrieve a list of databases or a single database within a catalog by providing its name.

```
my_catalog: Catalog
catalog_dbs: List[Database] = my_catalog.databases
my_db: Database = my_catalog.database("my_db")
```

Each Database object has several properties that can provide information about the database.

```
my_db.name  
my_db.catalog_id  
my_db.location_uri  
my_db.project_id  
my_db.domain_id
```

Tables

You can also retrieve a list of tables or a specific table within a Database.

```
my_db_tables: List[Table] = my_db.tables  
my_table: Table = my_db.table("my_table")
```

Each Table object has several properties that can provide information about the table.

```
my_table.name  
my_table.database_name  
my_table.catalog_id  
my_table.location
```

You can also retrieve a list of the columns within a table. Column contains the column name and the data type of the column.

```
my_table_columns: List[Column] = my_table.columns  
col_0: Column = my_table_columns[0]  
col_0.name  
col_0.type
```

Execution APIs

Execution APIs provide you the ability to start an execution to run a notebook headlessly within the same user space or on remote compute.

Local Execution APIs

Use the following APIs to start, stop, get, or list executions within the user's space.

StartExecution

You can start a notebook execution headlessly within the same user space.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(overrides={
    "execution": {
        "local": True,
    }
})
sagemaker_studio_api = SageMakerStudioAPI(config)

result = sagemaker_studio_api.execution_client.start_execution(
    execution_name="my-execution",
    input_config={"notebook_config": {
        "input_path": "src/folder2/test.ipynb"}},
    execution_type="NOTEBOOK",
    output_config={"notebook_config": {
        "output_formats": ["NOTEBOOK", "HTML"]
    }}
)
print(result)
```

GetExecution

You can retrieve details about a local execution using the GetExecution API.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(region="us-west-2", overrides={
    "execution": {
        "local": True,
```

```
        }
    })
sagemaker_studio_api = SageMakerStudioAPI(config)

get_response =
    sagemaker_studio_api.execution_client.get_execution(execution_id="asdf-3b998be2-02dd-42af-8802
print(get_response)
```

ListExecutions

You can use the `ListExecutions` API to list all the executions that ran in the user's space.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(region="us-west-2", overrides={
    "execution": {
        "local": True,
    }
})
sagemaker_studio_api = SageMakerStudioAPI(config)

list_executions_response =
    sagemaker_studio_api.execution_client.list_executions(status="COMPLETED")
print(list_executions_response)
```

StopExecution

You can use the `StopExecution` API to stop an execution that's running in the user space.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(region="us-west-2", overrides={
    "execution": {
        "local": True,
    }
})
```

```
sagemaker_studio_api = SageMakerStudioAPI(config)

stop_response =
    sagemaker_studio_api.execution_client.stop_execution(execution_id="asdf-3b998be2-02dd-42af-880
print(stop_response)
```

Remote Execution APIs

Use the following APIs to start, stop, get, or list executions running on remote compute.

StartExecution

You can start a notebook execution headlessly on a remote compute specified in the StartExecution request.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(region="us-west-2")
sagemaker_studio_api = SageMakerStudioAPI(config)

result = sagemaker_studio_api.execution_client.start_execution(
    execution_name="my-execution",
    execution_type="NOTEBOOK",
    input_config={"notebook_config": {"input_path": "src/folder2/test.ipynb"}},
    output_config={"notebook_config": {"output_formats": ["NOTEBOOK", "HTML"]}},
    termination_condition={"max_runtime_in_seconds": 9000},
    compute={
        "instance_type": "ml.c5.xlarge",
        "image_details": {
            # provide either ecr_uri or (image_name and image_version)
            "image_name": "sagemaker-distribution-embargoed-loadtest",
            "image_version": "2.2",
            "ecr_uri": "123456123456.dkr.ecr.us-west-2.amazonaws.com/ImageName:latest",
        }
    }
)
print(result)
```

GetExecution

You can retrieve details about an execution running on remote compute using the GetExecution API.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(region="us-west-2")
sagemaker_studio_api = SageMakerStudioAPI(config)

get_response =
    sagemaker_studio_api.execution_client.get_execution(execution_id="asdf-3b998be2-02dd-42af-8802
print(get_response)
```

ListExecutions

You can use the ListExecutions API to list all the headless executions that ran on remote compute.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig

config = ClientConfig(region="us-west-2")
sagemaker_studio_api = SageMakerStudioAPI(config)

list_executions_response =
    sagemaker_studio_api.execution_client.list_executions(status="COMPLETED")
print(list_executions_response)
```

StopExecution

You can use the StopExecution API to stop an execution that's running on remote compute.

```
from sagemaker_studio.sagemaker_studio_api import SageMakerStudioAPI
from sagemaker_studio import ClientConfig
```

```
config = ClientConfig(region="us-west-2")
sagemaker_studio_api = SageMakerStudioAPI(config)

stop_response =
    sagemaker_studio_api.execution_client.stop_execution(execution_id="asdf-3b998be2-02dd-42af-880
print(stop_response)
```

IDE spaces in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio provides compute spaces for integrated development environments (IDEs) that you can use to code and develop your resources. When you create and use these IDEs in a Amazon SageMaker Unified Studio project, you have access to all the data in that project as well and can share coding work with other project members.

There are two IDE applications available in Amazon SageMaker Unified Studio: JupyterLab and Code Editor. A JupyterLab space is created in your project by default, and you can create additional spaces as desired.

Creating a new space

To create a space, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project you want to create a space in. You can do this by choosing **Browse all projects** from the center menu.
3. In the **Build** menu, choose **Spaces** to navigate to the **Spaces** tab on the **Compute** page.
4. Choose **Create space**.
5. Under **Name**, enter a name for the space. This name must be unique. No spaces in the project can use the same name.
6. Under **Application**, choose the IDE application you want to use with the space.
7. Under **Instance**, choose an Amazon EC2 instance that is most compatible with your use case.

 **Note**

If you use a GPU instance type when configuring your Code Editor application, you must also use a GPU-based image. Within a space, your data is stored in an Amazon EBS volume that persists independently from the life of an instance. You won't lose your data when you change instances.

8. Under **Image**, choose the image you want to use.
9. Under **EBS space storage**, enter a value from 16 to 100 to choose a storage size.

10. (Optional) Under **Lifecycle configuration**, choose the kind of lifecycle you want for the space.
11. Under **Idle time**, enter the amount of time before the space will time out and stop running.
12. (Optional) Under **Attach custom file system**, attach a file system.
13. Choose **Create and start space** to create the space and have it start running.

It might take up to 5 minutes for Amazon SageMaker Unified Studio to finish creating the space. After the space is created, you can view it in the list on the **Spaces** tab and choose **Open** to launch the IDE in your browser.

You can then perform various actions using the three-dot **Actions** menu on the **Spaces** tab in Amazon SageMaker Unified Studio. You can edit the space, start or stop running the application, delete the space, or view information about the space.

Editing a space

Project members can edit a space at any time to change certain configurations. Before editing a space, make sure that you have saved any changes to your work in that space.

 **Note**

A space's name and IDE application cannot be changed.

To edit configurations for a space, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project you want to edit a space in. You can do this by choosing **Browse all projects** from the center menu.
3. In the **Build** menu, choose **Spaces** to navigate to the **Spaces** tab on the **Compute** page.
4. Choose the three-dot **Actions** menu next to the space you want to edit. Then choose **Configure space**.
5. Make the desired changes, then choose **Save and restart** if the space is running, or **Save and start** if the space is stopped.

Note

Saving changes to a running space requires a restart, and any unsaved changes will be lost.

Deleting a space

Project members can choose to delete a space in Amazon SageMaker Unified Studio. Deleting a space is permanent and cannot be undone. Before deleting a space, make sure that you have saved all your data and that you are sure that this is the action you would like to take.

Note

You cannot delete the default JupyterLab space in Amazon SageMaker Unified Studio.

To delete a space, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project you want to delete a space in. You can do this by choosing **Browse all projects** from the center menu.
3. In the **Build** menu, choose **Spaces** to navigate to the **Spaces** tab on the **Compute** page.
4. Choose the three-dot **Actions** menu next to the space you want to delete. Then choose **Delete**.
5. Review the information in the window provided, then type **confirm** where indicated.
6. Choose **Delete this space**.

Using the coding assistant

The Amazon SageMaker Unified Studio is integrated with Amazon Q. Amazon Q Developer is a coding assistant that can chat about code, provide inline code completions, or generate net new code.

For more information about Amazon Q Developer, see [What is Amazon Q Developer](#) in the Amazon Q Developer User Guide.

To use the Amazon Q Developer model for chat:

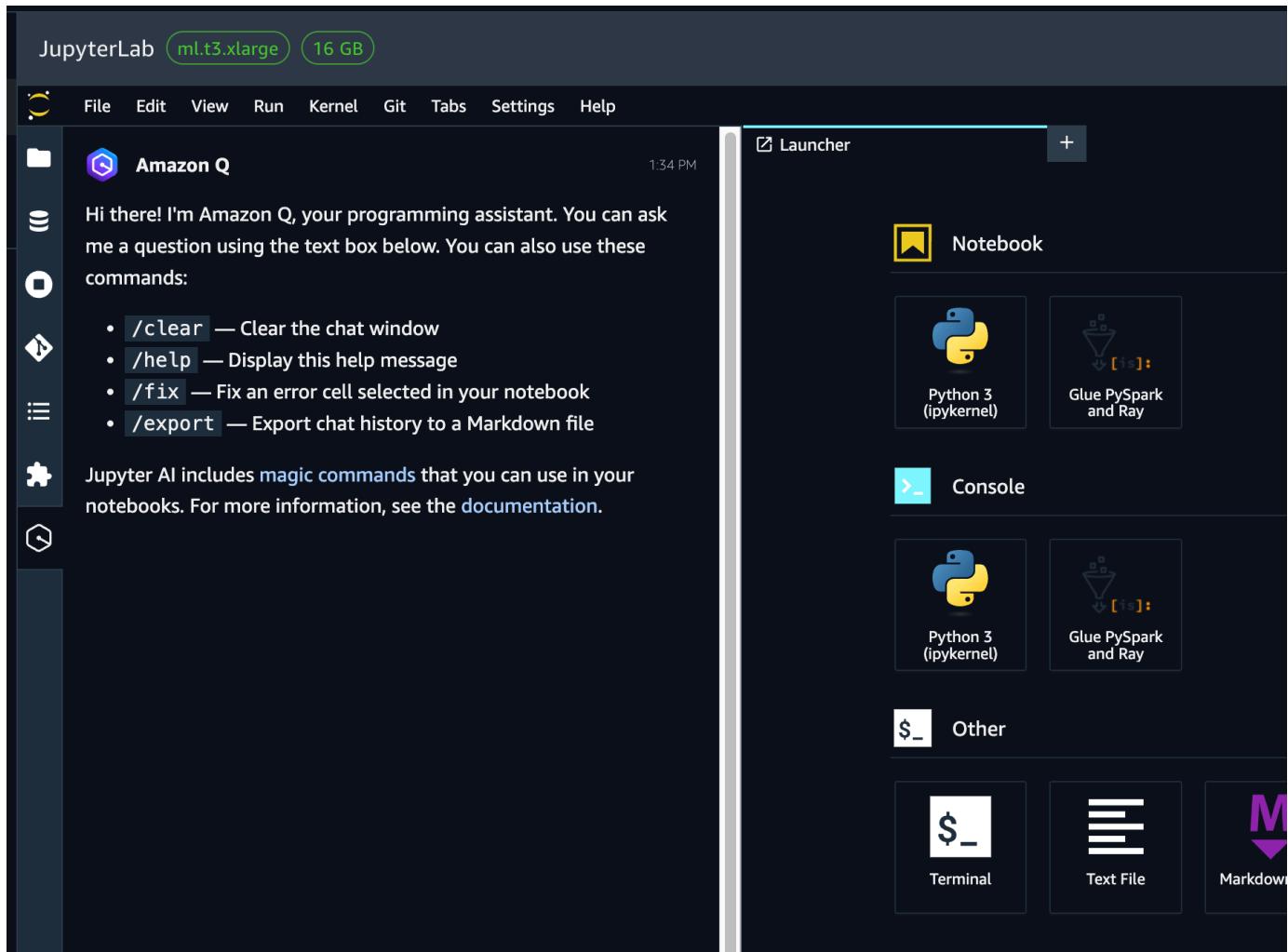
1. Ensure your admin must has subscribed to Amazon Q Developer and added Amazon Q Developer as an application to your domain in the Amazon Q Developer console, as described in the Amazon Amazon SageMaker Unified Studio Administrator Guide.

 **Note**

When you enable Amazon Q, you can pick between either the free or paid tiers of the service. JupyterLab supports both the free and paid tiers, and Code Editor supports the free tier.

When using the free tier, request limits are shared at the account level, meaning that one customer can potentially use up all requests. The pro tier of Amazon Q is charged at the user level, with limits set at the user level as well. The pro tier also lets you manage users and policies with enterprise access control.

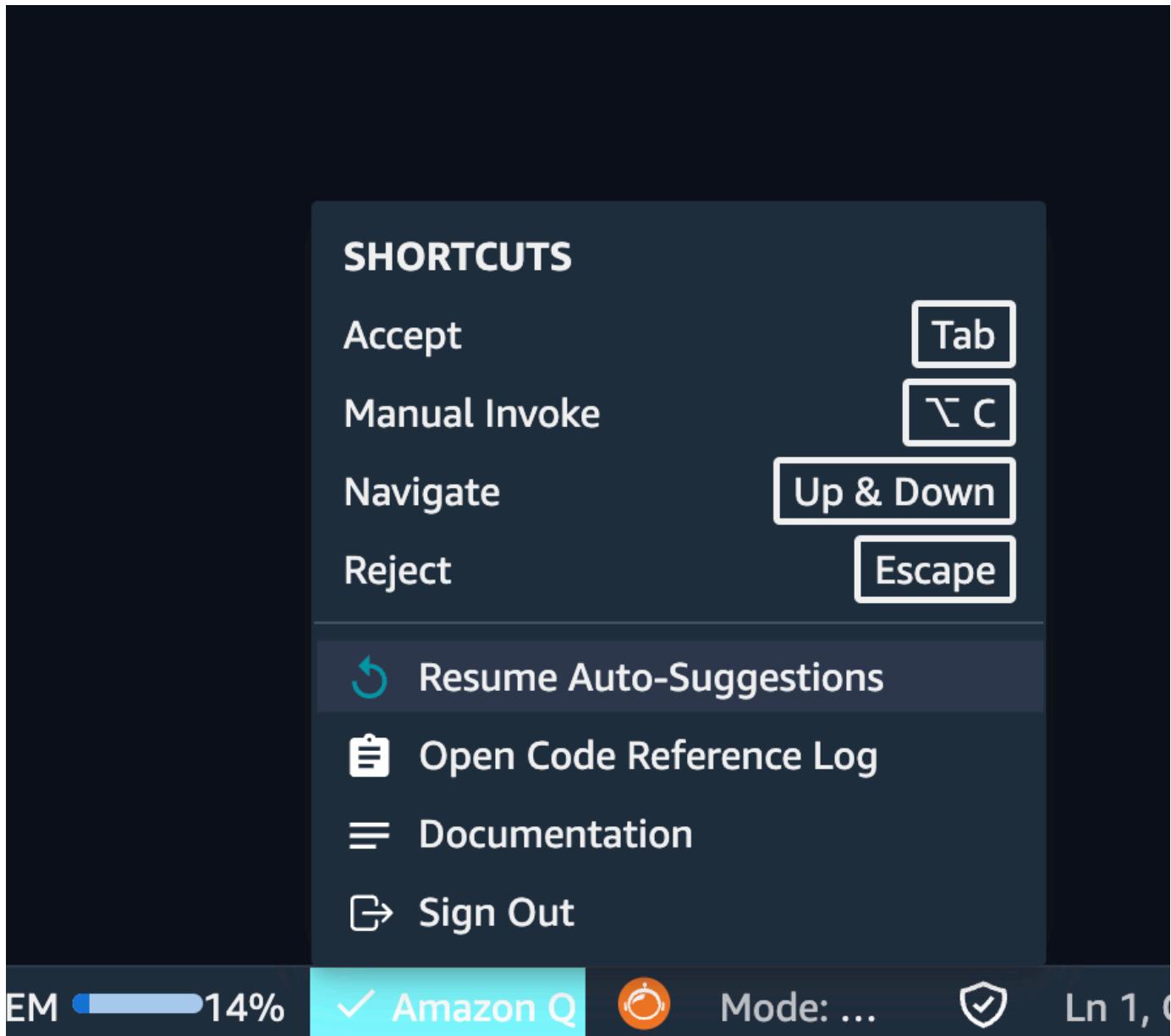
2. After adding Amazon Q Developer , you can access the chat interface by navigating to the JupyterLab or Code Editor experience and choosing the chat icon in the left navigation panel of your notebook in Amazon SageMaker Unified Studio.



3. You are now able to see code completions powered by Amazon Q Developer in your notebook. Amazon Q Developer makes code recommendations automatically as you write your code, based on your existing code and comments. For more information about how inline suggestions work in Amazon Q Developer , see [Generating inline suggestions](#) in the Amazon Q Developer User Guide.

Amazon Q Developer provides automatic suggestions for your code by default. To pause or resume automatic suggestions:

- Choose "Amazon Q" from the navigation bar at the bottom of the JupyterLab or Code Editor IDE. Then choose Pause Auto-Suggestions or Resume Auto-Suggestions, as desired.



If you want to opt out of Amazon Q data sharing, see the [opt-out section of the Amazon Q developer guide](#).

Using the JupyterLab IDE in Amazon SageMaker Unified Studio

The JupyterLab page of Amazon SageMaker Unified Studio provides a JupyterLab interactive development environment (IDE) for you to use as you perform data integration, analytics, or

machine learning in your projects. Amazon SageMaker Unified Studio notebooks are powered by JupyterLab spaces.

By default, the JupyterLab application comes with the Amazon SageMaker Distribution image. The distribution image includes popular packages such as the following:

- PyTorch
- TensorFlow
- Keras
- NumPy
- Pandas
- Scikit-learn

Amazon SageMaker Unified Studio includes a sample notebook that you can use to get started. You can also choose to create new notebooks for your business use cases.

Amazon SageMaker Unified Studio notebooks include the following key features:

- Manage configurations to scale the instance vertically if the job being submitted demands it.
- Access metadata to find out information such as the path to the Amazon S3 bucket where data is being stored.
- Perform Git operations for version control.
- Use Amazon Q chat functionality to ask questions and generate code using prompts.
- Perform code completion using Amazon Q Developer.

 **Note**

The JupyterLab IDE has an idle shutdown feature that shuts down the IDE after it has been idle for 60 minutes. This means that if both the IDE kernel and terminal have been unused for an hour, the IDE stops running. In order to start using the IDE again after idle shutdown, you would need to navigate to the JupyterLab page again and click on the Start button to restart the kernel in the JupyterLab IDE.

Managing configurations

You can edit your JupyterLab configurations on the JupyterLab page by choosing **Configure** in the top right corner. A popup appears where you can change the instance type. You can also increase the EBS volume up to 16 GB if allowed by your admin.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Expand the **Build** menu in the top navigation, then choose **JupyterLab**.
3. Choose the **Configure** button in the top right corner of the page. A popup appears where you can change the instance type and increase the EBS volume.
4. Specify the instance type and EBS volume that you want for testing.
 - NOTE: After you increase the EBS volume, you cannot decrease it.

Configuring Spark compute

Amazon SageMaker Unified Studio provides a set of Jupyter magic commands. Magic commands, or magics, enhance the functionality of the IPython environment. For more information about the magics that Amazon SageMaker Unified Studio provides, run `%help` in a notebook.

Compute-specific configurations can be set by using the `%%configure` Jupyter magic. The `%%configure` magic takes a JSON-formatted dictionary. To use `%%configure` magic, specify the compute name in the argument `-n`. Including `-f` will restart the session to forcefully apply the new configuration. Otherwise, this configuration will apply when the next session starts.

For example: `%%configure -n compute_name -f`.

Library management

You can use the library management widget in JupyterLab to manage the library installations and configurations in your notebook.

To navigate to the library management of a notebook in Amazon SageMaker Unified Studio, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

2. Navigate to a project. You can do this by choosing **Browse all projects** from the center menu and then selecting a project, or by creating a new project.
3. From the **Build** menu, choose **JupyterLab**.
4. Navigate to a notebook or create a new one by selecting **File > New > Notebook**.
5. Choose the library management icon from the notebook navigation bar.



The following library configurations are available:

Jar

- **Maven artifacts**
- **S3 paths**
- **Disk location paths**
- **Other paths**

Python

- **Conda packages**
- **PyPI packages**
- **S3 paths**
- **Disk location paths**
- **Other paths**

Adding JupyterLab library configurations

1. Navigate to the JupyterLab library management page.
2. Select the configuration method you would like to add from the left navigation of the library management page.
3. Choose **Add**.
4. Input the URL, package name, coordinates, or other information as the fields indicate.
5. In the left navigation of the library management page, check the box **Apply the change to JupyterLab**.
6. Choose **Save all changes**.

Compute-specific configuration

Amazon SageMaker Unified Studio provides a set of Jupyter magic commands. Magic commands, or magics, enhance the functionality of the IPython environment. For more information about the magics that Amazon SageMaker Unified Studio provides, run `%help` in a notebook.

Compute-specific configurations can be configured by `%%configure` Jupyter magic. The `%configure` magic takes a json-formatted dictionary. To use `%%configure` magic, please specify the compute name in the argument `-n`. Include `-f` will restart the session to forcefully apply the new configuration, otherwise this configuration will apply only when next session starts.

Configure an EMR Spark session

When working with EMR on EC2 or EMR Serverless, `%%configure` command can be used to configure the Spark session creation parameters. Using `conf` settings, you can configure any Spark configuration that's mentioned in the configuration documentation for Apache Spark.

```
%configure -n compute_name -f
{
    "conf": {
        "spark.sql.shuffle.partitions": "36"
    }
}
```

Configure a Glue interactive session

Use the -- prefix for run arguments specified for Glue.

```
%configure -n project.spark.compatibility -f
{
    "--enable-auto-scaling": "true"
    "--enable-glue-datacatalog": "false"
}
```

For more information on job parameters, see [Job parameters](#).

You can update Spark configuration via %%configure when working with Glue with --conf in configure magic. You can configure any Spark configuration that's mentioned in the configuration documentation for Apache Spark.

```
%configure -n project.spark.compatibility -f
{
    "--conf": "spark.sql.shuffle.partitions=36"
}
```

Accessing metadata

You can view metadata for your project in the notebook terminal within Amazon SageMaker Unified Studio. This shows you information such as the ProjectS3Path, which is the Amazon S3 bucket where your project data is stored. The project metadata is written to a file named resource-metadata.json in the folder /opt/ml/metadata/. You can get the metadata by opening a terminal from within the notebook.

1. Navigate to the Code page within the project you want to view metadata for.
2. Choose File > New > Terminal.
3. Enter in the following command:

```
cat /opt/ml/metadata/resource-metadata.json
```

The metadata file information then appears in the terminal window.

Performing Git operations

The JupyterLab IDE in Amazon SageMaker Unified Studio is configured with Git and initialized with the project repository when a project is created.

To access Git operations in the Amazon SageMaker Unified Studio management console, navigate to the Code page of your project, then choose the Git button in the JupyterLab IDE left panel as shown in the image below.

This opens a panel where you can view commit history and perform Git operations. You can use this Git extension to commit and push files back to the project repository, switch your working branch or create a new one, and manage tags.

To fetch notebooks committed by other users, do a pull from the project repository.

Note

When you create and enable a connection for Git access and the user accesses this connection in the JupyterLab IDE in Amazon SageMaker Unified Studio, the repository is cloned. In other words, a local copy of the repository is created in the Amazon SageMaker Unified Studio project. If the administrator later disables or deletes this Git connection, the local repository remains in the user's IDE, but users can no longer push or pull files to or from it. For more information, see [Git connections in Amazon SageMaker Unified Studio](#).

Using the Amazon Q data integration in AWS Glue

Amazon SageMaker Unified Studio supports the [Amazon Q data integration](#) in AWS Glue. It helps data engineers and ETL developers create data integration jobs using natural language letting you automate aspects of code authoring.

When using the Amazon Q data integration in AWS Glue, in the Jupyter Lab IDE, you enter comments using natural language instructions, and then the PySpark kernel generates the code on your behalf. You can customize the generated code to meet your own needs.

1. Open a Python notebook, and ensure the kernel is configured to use a PySpark connection.

2. You can request a prompt response by adding a comment and then placing it in a prompt, which will start Amazon Q processing.
3. If the prompt is AWS Glue related, the data integration generates a AWS Glue job script using PySpark.
4. Alternatively, you can continue to use your default auto-completions from Amazon Q Developer. If a prompt isn't Glue related, Amazon Q Developer will use autocomplete instead.

Running SQL and Spark code

You can run code against multiple compute in one Jupyter notebook using different programming languages through the use of Jupyter cell magic %%pyspark, %%sql, %%scalaspark.

For example, to run pyspark code on Spark compute, you can run the following code:

```
%%pyspark compute_name
spark.createDataFrame([('Alice', 1)])
```

The following table represents the supported compute types of each magic:

magic	supported compute types
%%sql	Redshift, Athena, EMR on EC2, EMR Serverless, Glue Interactive Session
%%pyspark	EMR on EC2, EMR Serverless, Glue Interactive Session
%%scalaspark	EMR on EC2, EMR Serverless, Glue Interactive Session

The dropdown available at the top of active cells allows you to select the Connection and Compute type. If no selection is made the code in the cell will be run against the Compute hosting JupyterLab ("Local Python" / "project.python"). The option selected for Connection type dictates the Compute available. The selections dictate the magic code generated in the cell and determine where your code runs.

When a new cell is created, it will select the same connection and compute type as the previous cell automatically. To configure the dropdown, go to Settings → Settings editor → Connection magics settings.

Visualizing results

`%display` is a magic that customers can apply against any DataFrame to invoke a visualization for tabular data. Use the visualization to scroll through a DataFrame or results of a Redshift or Athena query.

There are four different views:

- **Table.** You can change the sampling method, sample size, and rows per page that are displayed.
- **Summary.** Each column in the summary tab has a button labeled with the column's name. Clicking on one of these buttons opens the a sub-tab in the column view in Tab 3 for the column that was clicked.
- **Column.** For each column selected in the column selector above, a sub-tab appears with more details about the contents of the column.
- **Plotting.** In the default plotting view you can change the graph type, axes, value types, and aggregation functions for plotting. By installing an optional supported third-party plotting library on the Jupyterlab space (pygwalker, ydata-profiling, or dataprep) and running the display magic you can visualize your data using the installed library.

Using the Code Editor IDE in Amazon SageMaker Unified Studio

Code Editor, based on Code-OSS, Visual Studio Code - Open Source, helps you write, test, debug, and run your analytics and machine learning code. It also supports integrated development environment (IDE) extensions available in the Open VSX Registry.

After creating a Code Editor space in Amazon SageMaker Unified Studio, you can access your Code Editor session directly through the browser. Within your Code Editor environment, you can do the following:

- Access artifacts from Amazon SageMaker Unified Studio
- Clone your GitHub repositories and commit changes
- Access the SageMaker Python SDK

You can return to Amazon SageMaker Unified Studio to review any assets created in your Code Editor environment such as experiments, pipelines, or training jobs.

Performing Git operations in Code Editor

The Code Editor IDE in Amazon SageMaker Unified Studio is configured with Git and initialized with the project repository when a Code Editor space is created.

You can use Git operations after launching your Code Editor space. To launch your project, choose **Open** next to the space you want to open in the **Spaces** tab of your project.

When Code Editor is open, choose the **Source Control** icon in the left navigation. You can use this window to perform Git operations such as committing and pushing files back to the project repository, switching your working branch or creating a new one, and managing tags.

To fetch notebooks committed by other users, do a pull from the project repository.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to your project.
3. In the **Build** menu, choose **Spaces**.
4. Choose **Open** next to the Code Editor space you want to open.
5. In the Source Control window of your Code Editor space, choose the three-dot menu. Then choose **Pull**.

Note

When you create and enable a connection for Git access and the user accesses this connection in the Code Editor IDE in Amazon SageMaker Unified Studio, the repository is cloned. In other words, a local copy of the repository is created in the Amazon SageMaker Unified Studio project. If the administrator later disables or deletes this Git connection, the local repository remains in the user's IDE, but users can no longer push or pull files to or from it. For more information, see [Git connections in Amazon SageMaker Unified Studio](#).

Checking the version of Code Editor in Amazon SageMaker Unified Studio

The following steps show how to check the version of your Code Editor application.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to your project.
3. In the **Build** menu, choose **Spaces**.
4. Choose **Open** next to the Code Editor space you want to open.
5. In the upper-left corner of the Code Editor UI, choose the three-bar menu button.
6. Choose **Help**. Then choose **About**.

Connections and extensions in Code Editor

Code Editor supports IDE connections to AWS services as well as extensions available in the [Open VSX Registry](#).

Code Editor environments are integrated with the AWS Toolkit for VS Code to add connections to AWS services. Within your Code Editor environment, you can add connections to [AWS Explorer](#) to view, modify, and deploy AWS resources in Amazon S3, CloudWatch, and more.

Code Editor supports IDE extensions available in the [Open VSX Registry](#). To get started with extensions in your Code Editor environment, choose the Extensions icon in the left navigation pane. Here, you can configure connections to AWS by installing the AWS Toolkit. For more information, see [Installing the AWS Toolkit for Visual Studio Code](#).

In the search bar, you can search directly for additional extensions through the [Open VSX Registry](#), such as the AWS Toolkit, Jupyter, Python, and more.

Data

Data in Amazon SageMaker Unified Studio includes data in projects of which you are a member and data that you can discover and subscribe to from other projects.

The **Data** page in Amazon SageMaker Unified Studio displays a data browser in which you can explore datasets, files, and artifacts that you connect to your project. Projects configured with certain profiles contain an Amazon SageMaker Lakehouse for accessing data within your project, as well as a default Amazon Redshift connection and an Amazon S3 bucket. You can add data to the project on the **Data** page by uploading data from your local desktop or by gaining access to existing data sources and then adding a connection to them in your Amazon SageMaker Unified Studio project. For more information about using the Data page and Amazon SageMaker Lakehouse, see [Amazon SageMaker Lakehouse](#) and [the section called "S3 data"](#).

You can also connect to AWS Glue and Amazon Redshift data sources from within your project catalog. The project catalog contains your data as data products and assets with metadata. When you want to share your data with other projects in the domain, publish the data from your project catalog into the Amazon SageMaker catalog. If you want to create more detailed access control for your data before allowing other users to subscribe to it, you can configure fine-grained access control. For more information, see [the section called "Data inventory and publishing"](#) and [the section called "Fine-grained access control to data"](#).

The Amazon SageMaker catalog contains business glossaries and metadata forms. If you have been granted access through the authorization policies, you can create business glossaries and metadata forms. For more information, see [Domain units and authorization policies in Amazon SageMaker Unified Studio](#) and [the section called "Data catalog"](#).

You can use the Amazon SageMaker catalog to discover and subscribe to assets and data products. For more information, see [the section called "Data discovery, subscription, and consumption"](#).

Amazon S3 data in Amazon SageMaker Unified Studio

You can bring in Amazon S3 data to your project and access it on the **Data** page in Amazon SageMaker Unified Studio.

Adding Amazon S3 data

To bring in Amazon S3 data to your project, you must first gain access to the data and then add the data to your project. You can gain access to the data by using the project role or an access role.

Note

If you are using a bucket in a different account than the account that contains the project tooling environment, you must use an access role to gain access to the data.

Prerequisite option 1: Gain access using the project role

Work with your admin to complete the following steps:

1. Retrieve the project role ARN and send it to your admin.
 - a. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
 - b. Navigate to the project that you want to add Amazon S3 data to. You can do this by choosing **Browse all projects** from the center menu, and then selecting the name of the project.
 - c. On the **Project overview** page, copy the project role ARN.
2. The admin then must go to the Amazon S3 console and add a CORS policy to the bucket that you want to access in your project.
 - a. Navigate to the Amazon S3 console.
 - b. Navigate to the bucket you want to grant access to.
 - c. On the **Permissions** tab, under **Cross-origin resource sharing (CORS)**, choose **Edit**.
 - d. Enter in the new CORS policy, then choose **Save changes**.

```
[  
 {  
     "AllowedHeaders": [  
         "*"  
     ],  
     "AllowedMethods": [  
         "PUT",  
         "POST",  
         "DELETE",  
         "GET"  
     ]  
 }]
```

```
        "PUT",
        "GET",
        "POST",
        "DELETE",
        "HEAD"
    ],
    "AllowedOrigins": [
        "domainUrl" // example: https://dzd_abcdefg1234567.sagemaker.us-
east-1.on.aws
    ],
    "ExposeHeaders": [
        "x-amz-version-id"
    ]
}
```

- e. Choose the name of an object to view its details. On the **Properties** tab, note the resource name ARN and the S3 URI. You will need to use these later.
3. The admin then must go to the IAM console and update the project role.
 - a. Navigate to the IAM console.
 - b. On the **Roles** page, search for the project role using the last string in the project role ARN, for example: datazone_usr_role_1a2b3c45de6789_abcd1efghij2kl.
 - c. Select the project role to navigate to the project role details.
 - d. Under the **Permissions** tab, choose **Add permissions**, then choose **Create inline policy**.
 - e. Use the JSON editor to create a policy so that the project has access to an Amazon S3 location, using the Amazon S3 resource ARN that you noted in step 2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "S3AdditionalBucketPermissions",
            "Effect": "Allow",
            "Action": [
                "s3>ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::dzd_abcdefg1234567"
            ]
        }
    ]
}
```

```
        "arn:aws:s3:::bucketName"  
    ]  
,  
{  
    "Sid": "S3AdditionalObjectPermissions",  
    "Effect": "Allow",  
    "Action": [  
        "s3:GetObject*",  
        "s3:PutObject"  
    ],  
    "Resource": [  
        "arn:aws:s3:::bucketName/key/*"  
    ]  
}  
]  
}
```

f. Choose **Next**

g. Enter a name for the policy, then choose **Create policy**.

Prerequisite option 2: Gain access using an access role

Work with your admin to complete the following steps:

1. Retrieve the project role ARN and the project ID and send them to your admin.
 - a. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
 - b. Navigate to the project that you want to add Amazon S3 data to. You can do this by choosing **Browse all projects** from the center menu, and then selecting the name of the project.
 - c. On the **Project overview** page, copy the project role ARN and the project ID.
2. The admin then must go to the Amazon S3 console and add a CORS policy to the bucket that you want to access in your project.
 - a. Navigate to the Amazon S3 console.
 - b. Navigate to the bucket you want to grant access to.
 - c. On the **Permissions** tab, under **Cross-origin resource sharing (CORS)**, choose **Edit**.
 - d. Enter in the new CORS policy, then choose **Save changes**.

```
[  
  {  
    "AllowedHeaders": [  
      "*"  
    ],  
    "AllowedMethods": [  
      "PUT",  
      "GET",  
      "POST",  
      "DELETE",  
      "HEAD"  
    ],  
    "AllowedOrigins": [  
      "domainUrl" // example: https://dzd_abcdefg1234567.sagemaker.us-  
east-1.on.aws  
    ],  
    "ExposeHeaders": [  
      "x-amz-version-id"  
    ]  
  }  
]
```

- e. Choose the name of an object to view its details. On the **Properties** tab, note the resource name ARN and the S3 URI. You will need to use these later.
3. The admin then must go to the IAM console and create an access role.
 - a. Navigate to the IAM console.
 - b. On the **Roles** page, choose **Create role**.
 - c. Under **Trusted entity type**, choose **AWS service**.
 - d. Under **Use case**, select **S3** from the dropdown menu, then choose the **S3** button.
 - e. Choose **Next**.
 - f. Search for and select the **AmazonS3FullAccess** policy, then choose **Next**.
 - g. Enter a name for the role, then choose **Create role**.
 - h. Select the access role from the list on the **Roles** page.
 - i. Under **Trust relationships** choose **Edit trust policy**.
 - j. Edit the trust policy to include the project ID and project ARN.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "project-role-arn"  
            },  
            "Action": "sts:AssumeRole",  
            "Condition": {  
                "StringEquals": {  
                    "sts:ExternalId": "project-id"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "project-role-arn"  
            },  
            "Action": [  
                "sts:SetSourceIdentity"  
            ],  
            "Condition": {  
                "StringLike": {  
                    "sts:SourceIdentity": "${aws:PrincipalTag/datazone:userId}"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "project-role-arn"  
            },  
            "Action": "sts:TagSession",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/AmazonDataZoneProject": "project-id",  
                    "aws:RequestTag/AmazonDataZoneDomain": "domain-id"  
                }  
            }  
        }  
    ]  
}
```

```
    ]  
}
```

- k. Optional: If the bucket is in a different account than the the access role, ensure cross-account bucket permissions are set by adding a bucket policy that grants cross-account permissions to the access role. For example:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "S3AdditionalBucketPermissions",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "access-role-arn"  
            },  
            "Action": [  
                "s3>ListBucket",  
                "s3:GetBucketLocation"  
            ],  
            "Resource": [  
                "arn:aws:s3:::bucketName"  
            ]  
        },  
        {  
            "Sid": "S3AdditionalObjectPermissions",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "access-role-arn"  
            },  
            "Action": [  
                "s3GetObject*",  
                "s3:PutObject"  
            ],  
            "Resource": [  
                "arn:aws:s3:::bucketName/key/*"  
            ]  
        }  
    ]  
}
```

- I. Choose **Update policy**.

Add the data to your project

When your admin has granted your project access to the Amazon S3 resources, you can add them to your project.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that you want to add Amazon S3 data to.
3. On the **Data** page, choose the plus icon +.
4. Select **Add S3 location**, then choose **Next**.
5. Enter a name for the location path.
6. (Optional) Add a description of the location path.
7. Use the S3 URI and Region provided by your admin.
8. If your admin has granted you access using an access role instead of the project role, enter the access role ARN from your admin.
9. Choose **Add S3 location**.

The S3 data is then accessible within your project in the left navigation on the **Data** page.

Amazon SageMaker Unified Studio data catalog

You can use the Amazon SageMaker Unified Studio business data catalog to catalog data across your organization with business context and thus enable everyone in your organization to find and understand data quickly.

In order to use Amazon SageMaker Unified Studio to catalog your data, you must first bring your data (assets) as inventory of your project in Amazon SageMaker Unified Studio. Creating inventory for a project makes the assets discoverable only to that project's members. Project inventory assets are not available to all domain users in search/browse unless explicitly published.

After creating a project inventory, data owners can curate their inventory assets with the required business metadata by adding or updating business names (asset and schema), descriptions (asset and schema), read me, glossary terms (asset and schema), and metadata forms.

The next step of using Amazon SageMaker Unified Studio to catalog your data, is to make your project's inventory assets discoverable by the domain users. You can do this by publishing the inventory assets to the Amazon SageMaker Unified Studio catalog. Only the latest version of the inventory asset can be published to the catalog and only the latest published version is active in the discovery catalog. If an inventory asset is updated after it's been published into the Amazon SageMaker Unified Studio catalog, you must explicitly publish it again in order for the latest version to be in the discovery catalog.

For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Topics

- [Create a business glossary in Amazon SageMaker Unified Studio](#)
- [Edit a business glossary in Amazon SageMaker Unified Studio](#)
- [Delete a business glossary in Amazon SageMaker Unified Studio](#)
- [Create a term in a glossary in Amazon SageMaker Unified Studio](#)
- [Edit a term in a glossary in Amazon SageMaker Unified Studio](#)
- [Delete a term in a glossary in Amazon SageMaker Unified Studio](#)
- [Create a metadata form in Amazon SageMaker Unified Studio](#)
- [Edit a metadata form in Amazon SageMaker Unified Studio](#)
- [Delete a metadata form in Amazon SageMaker Unified Studio](#)
- [Create a field in a metadata form in Amazon SageMaker Unified Studio](#)
- [Edit a field in a metadata form in Amazon SageMaker Unified Studio](#)
- [Delete a field in a metadata form in Amazon SageMaker Unified Studio](#)

Create a business glossary in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms (words) that may be associated with assets (data). It provides appropriate vocabularies with a list of business terms and their definitions for business users to ensure the same definitions are used across the organization when analyzing data. Business glossaries are created in the catalog domain and can be applied to assets and columns to help understand key characteristics of that asset or column. One or more glossary terms can be applied. A business glossary can be a flat list of terms where any term in the business glossary can be associated with a sublist of other terms. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or

delete a glossary in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project with the right permissions for that domain.

To create a glossary, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**, and then choose **Create glossary**.
4. Specify a name, description, and owning project for the glossary and then choose **Create glossary**.
5. Enable the new glossary by choosing the **Enabled** toggle.

To disable or enable a business glossary, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**.
4. Select the business glossary that you want to disable or enable.
5. On the glossary details page, locate the **Enabled** toggle and use it to enable or disable your selected glossary.

 **Note**

Disabling a glossary also disables all the terms that it contains.

Edit a business glossary in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms (words) that may be associated with assets (data). It provides appropriate vocabularies with a list of business terms and their definitions for business users to ensure the same definitions are used across the organization when analyzing data. Business glossaries are created in the catalog domain and can be applied to assets and columns to help understand key characteristics of that asset or column. One or more glossary terms can be applied. A business glossary can be a flat list of terms

where any term in the business glossary can be associated with a sublist of other terms. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To edit a glossary in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project with the right permissions for that domain.

To edit a business glossary, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**.
4. Select the business glossary that you want to edit.
5. On the glossary details page, expand **Actions** and then choose **Edit business glossary** to edit the glossary.
6. Make your updates to the name and description, and then choose **Update glossary**.

Delete a business glossary in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms (words) that may be associated with assets (data). It provides appropriate vocabularies with a list of business terms and their definitions for business users to ensure the same definitions are used across the organization when analyzing data. Business glossaries are created in the catalog domain and can be applied to assets and columns to help understand key characteristics of that asset or column. One or more glossary terms can be applied. A business glossary can be a flat list of terms where any term in the business glossary can be associated with a sublist of other terms. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To delete a glossary in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project with the right permissions for that domain.

To delete a business glossary, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**.
4. Select the business glossary that you want to delete.

5. On the glossary details page, make sure the **Enabled** toggle is off, so that the glossary is disabled.
6. Expand **Actions** and then choose **Delete** to delete the glossary.

 **Note**

You must delete all existing terms in the glossary before you can delete the glossary.

7. Confirm the deletion of the glossary by choosing **Delete**.

Create a term in a glossary in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms that may be associated with assets (data). For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete terms in a glossary in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project with the right permissions for that domain.

In Amazon SageMaker Unified Studio, business glossary terms can have close descriptions. To set the context of a particular term, you can specify relationships among terms. When you define a relationship for a term, it is automatically added to the definition of the related term. The glossary term relationships available in Amazon SageMaker Unified Studio include the following:

- **Is a Type of** - indicates that the current term is a type of the identified term. Indicates that the identified term is a parent to the current term.
- **Has Types** - indicates that the current term is a generic term for the indicated specific term or terms. This relationship can denote child terms for the generic term.

To create a new term, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**.
4. Select the glossary where you want to create the new term.
5. Choose **Create term**.

6. Specify a name and description for the term and then choose **Create term**.
7. Enable the new term by choosing the **Enabled** toggle.
8. To add a **Readme**, select the name of the term to navigate to the term details page. Then choose **Create readme** to add some additional information about this glossary.
9. To add relationships, complete the following steps:
 - a. Select the name of the term to navigate to the term details page.
 - b. If this is the first relationship added to the term, under **Terms relationships**, choose **Add terms**. If there are other terms relationships listed, under **Term Relationships**, choose **Edit**, and then choose **Add terms**.
 - c. In the dialog, choose the relationship and the terms you want to relate.
 - d. Choose **Add terms** to add the selected terms to the appropriate relationship type. This relationship is also added to all the terms you made related.

Edit a term in a glossary in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms that may be associated with assets (data). For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete terms in a glossary in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project with the right permissions for that domain.

In Amazon SageMaker Unified Studio, business glossary terms can have close descriptions. To set the context of a particular term, you can specify relationships among terms. When you define a relationship for a term, it is automatically added to the definition of the related term. The glossary term relationships available in Amazon SageMaker Unified Studio include the following:

- **Is a Type of** - indicates that the current term is a type of the identified term. Indicates that the identified term is a parent to the current term.
- **Has Types** - indicates that the current term is a generic term for the indicated specific term or terms. This relationship can denote child terms for the generic term.

To edit a term in a glossary, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**.
4. Select the glossary that contains the term that you want to edit.
5. Choose the name of the term to navigate to the term details page.
6. On the term details page, expand **Actions** and then choose **Edit** to edit the term.
7. Make your updates to the name and description, and then choose **Update term**.

Delete a term in a glossary in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, a business glossary is a collection of business terms that may be associated with assets (data). For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete terms in a glossary in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project with the right permissions for that domain.

In Amazon SageMaker Unified Studio, business glossary terms can have close descriptions. To set the context of a particular term, you can specify relationships among terms. When you define a relationship for a term, it is automatically added to the definition of the related term. The glossary term relationships available in Amazon SageMaker Unified Studio include the following:

- **Is a Type of** - indicates that the current term is a type of the identified term. Indicates that the identified term is a parent to the current term.
- **Has Types** - indicates that the current term is a generic term for the indicated specific term or terms. This relationship can denote child terms for the generic term.

To delete a term in a glossary, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Glossaries**.
4. Select the glossary that contains the term that you want to delete.
5. Select the name of the term to navigate to the term details page.
6. Make sure that the **Enabled** toggle is off so that the term is disabled.
7. On the glossary term details page, expand **Actions** and then choose **Delete**.

8. Confirm the deletion of the term by choosing **Delete**.

Create a metadata form in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. They serve as extensible mechanisms for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve a mechanism to enforce consistency to all assets being published to the Amazon SageMaker Unified Studio catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete metadata forms in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project who has the right credentials.

To create a metadata form, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Metadata forms**.
4. Choose **Create metadata form**.
5. Specify the metadata form technical name, owning project, and optional display name and description.
6. Choose **Create metadata form**.

Edit a metadata form in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. They serve as extensible mechanisms for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve a mechanism to enforce consistency to all assets being published to the Amazon SageMaker Unified Studio catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information,

see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete metadata forms in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project who has the right credentials.

To edit a metadata form, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Metadata forms**.
4. Choose the name of the metadata form that you want to edit. This takes you to the metadata form details page.
5. On the metadata form details page, expand **Actions**, and then choose **Edit metadata form**.
6. Perform your updates to the name, description, and owning project.
7. Choose **Update form**.

Delete a metadata form in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. They serve as extensible mechanisms for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve a mechanism to enforce consistency to all assets being published to the Amazon SageMaker Unified Studio catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete metadata forms in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project who has the right credentials.

To delete a metadata form, complete the following steps:

 **Note**

Before you can delete a metadata form, you must remove it from all asset types or assets to which it is applied.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Metadata forms**.
4. Choose the name of the metadata form that you want to delete. This takes you to the metadata form details page.
5. If the metadata form that you want to delete is enabled, disable the metadata form by choosing the **Enabled** toggle.
6. On the metadata form's details page, expand **Actions**, and then choose **Delete**.
7. Confirm deletion by choosing **Delete**.

Create a field in a metadata form in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. They serve as extensible mechanisms for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve as a mechanism to enforce consistency to all assets being published to the Amazon SageMaker Unified Studio catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete fields in metadata forms in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project who has the right credentials.

To create a field in a metadata form, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Metadata forms**.
4. Choose the name of the metadata form that you want to add a field to. This takes you to the metadata form details page.
5. On the metadata form details page, choose **Create field**.

6. Specify the field name, description, type, and whether this is a required field. Depending on the field type, you might be able to configure additional selections.
7. Choose **Create field**.

Edit a field in a metadata form in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. They serve as extensible mechanisms for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve as mechanism to enforce consistency to all assets being published to the Amazon SageMaker Unified Studio catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete fields in metadata forms in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project who has the right credentials.

To edit a field in a metadata form, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Metadata forms**.
4. Choose the name of the metadata form that contains the field that you want to edit. This takes you to the metadata form details page.
5. On the metadata form details page, choose the field that you want to edit.
6. Expand the **Actions** menu, and then choose **Edit field**.
7. Make your updates to the field name, description, type, and whether it is a required field. Make updates to other selections if more are available with the selected field type.
8. Choose **Save**.

Delete a field in a metadata form in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. They serve as extensible mechanisms for

data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve as mechanism to enforce consistency to all assets being published to the Amazon SageMaker Unified Studio catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). To create, edit, or delete fields in metadata forms in your Amazon SageMaker Unified Studio domain, you must be a member of the owning project who has the right credentials.

To delete a field in a metadata form, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Metadata forms**.
4. Choose the name of the metadata form that contains the field that you want to delete. This takes you to the metadata form details page.
5. On the metadata form details page, choose the field that you want to delete.
6. Expand the **Actions** menu, and choose **Delete**.
7. Confirm deletion by choosing **Delete**.

Data products

In Amazon SageMaker Unified Studio, data producers can group data assets into well-defined, self-contained packages called *data products* that are tailored for specific business use cases. Using cohesive, business-aligned data products enhances both the publishing and the subscription processes.

Data consumers can identify interconnected data assets by searching for and finding them as a single unit. This approach reduces the time and effort required to find all relevant information and lowers the risk of missing important data. Also, data products simplify access to data with a single request by implementing a unified access model. This eliminates the need for multiple permissions, thereby speeding up the initiation of data analysis.

By cataloging assets as data products, data producers reduce administrative overhead by enabling metadata and access control management at the data product level, rather than individually. The

ability to surface these purpose-built grouped assets for consumption makes access governance and data utilization more efficient, ensuring it aligns with business goals and is easily accessible for its intended use. Data governance teams can monitor consumption rates for these data products, providing valuable insights into data literacy maturity. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Topics

- [Create new data products in Amazon SageMaker Unified Studio](#)
- [Publish data products in Amazon SageMaker Unified Studio](#)
- [Edit data products in Amazon SageMaker Unified Studio](#)
- [Unpublish data products in Amazon SageMaker Unified Studio](#)
- [Delete data products in Amazon SageMaker Unified Studio](#)
- [Subscribe to a data product in Amazon SageMaker Unified Studio](#)
- [Review a subscription request and grant a subscription to a data product in Amazon SageMaker Unified Studio](#)
- [Republish data products in Amazon SageMaker Unified Studio](#)

Create new data products in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can create a Amazon SageMaker Unified Studio data product.

To create a new data product complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project in which you'd like to create a data product. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.
4. Expand the **Create** menu and then choose **Create data product**.

5. In the **Create new data product** page, complete the following steps.
 - a. Specify the name and the description for the data product
 - b. Choose **Select assets** to add various assets to your data product.
 - c. In the **Select assets** pop-up window, choose **Choose** next to the assets that you want to add to this data product.
 - d. Choose **Choose** at the bottom of the pop-up window.
6. To complete creating the data product, choose **Create**.

Publish data products in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can publish an Amazon SageMaker Unified Studio data product.

To publish a data product complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data product that you want to publish. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.
4. Choose the **Inventory** tab, and then choose the **Data products** filter. This displays existing data products in the project inventory.
5. Choose the data product that you want to publish. This opens the data product details page.
6. Choose **Publish**. Confirm the publishing of this data product by choosing **Publish data product**.

 **Note**

Any unpublished data assets that are in this data product will become published, but will only be available through this data product.

Edit data products in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can edit an Amazon SageMaker Unified Studio data product.

To edit a data product complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data product that you want to edit. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.
4. Choose the **Inventory** tab, and then choose the **Data products** filter. This displays existing data products in the project inventory.
5. Choose the data product that you want to edit. As part of editing a data product, you can do the following:
 - Choose **Create README** to add a README that will help users understand the page better.
 - Choose **Add terms** to add glossary terms. Make your selections of glossary terms in the **Add terms** window and then choose **Add terms**.
 - Choose **Add metadata form** and then select your form in the **Add metadata form** window and choose **Add**.
 - Expand **Actions**, choose **Edit**, make your edits to the name and description of the data product, and then choose **Update**.
 - On the **Assets** tab, remove one of the existing assets in the data product by choosing that asset, then expanding the three-dot action icon and choosing **Remove asset**. Confirm the asset removal by choosing **Remove** in the **Remove asset** pop-up window. When you republish, this asset will be removed from all subscribers to this data product.
 - On the **Assets** tab, add a new asset to the data product by choosing the **Add** button and then selecting one or more assets to be added to the data product.

Unpublish data products in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can unpublish an Amazon SageMaker Unified Studio data product.

To unpublish a data product complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data product that you want to unpublish. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.
4. Choose the **Inventory** tab, and then choose the **Data products** filter. This displays existing data products in the project inventory.
5. Choose the data product that you want to unpublish. This opens the data product details page.
6. Expand **Actions** and choose **Unpublish**. Confirm the unpublishing of this data product by choosing **Unpublish**.

Note

Unpublishing a data product has the following effects:

- This data product will no longer be available to view or to subscribe to.
- Any data assets that are only available through this data product will no longer be available.
- All active subscriptions to this data product will remain.
- Any individually published data assets will not be affected.

Delete data products in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can delete an Amazon SageMaker Unified Studio data product.

To delete a data product complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data product that you want to delete. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.
4. Choose the **Inventory** tab, and then choose the **Data products** filter. This displays existing data products in the project inventory.
5. Choose the data product that you want to delete.
6. Expand **Actions** and choose **Delete**. Confirm the deletion of this data product by typing **delete** in the text field and then choosing **Delete**.

 **Note**

Deleting a data product has the following effects:

- The data product will no longer be available to publish, view, or subscribe to.
- Any data assets that are only available through this data product will no longer be visible in the data catalog. They will not be deleted from your inventory assets.

Subscribe to a data product in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can subscribe to an Amazon SageMaker Unified Studio data product.

To subscribe to a data product, complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Data catalog**.
4. Choose **Browse data products**.
5. Find the data product to which you want to subscribe and then choose that data product.
6. On the data product details page, choose **Subscribe**.
7. Specify the project and the reason for requesting a subscription. Then choose **Request**.

When the owning project grants the subscription request, you will be subscribed to the data product.

Review a subscription request and grant a subscription to a data product in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

The owning project of the data product can review and grant the subscription to an Amazon SageMaker Unified Studio data product.

To review a subscription request and grant or reject a subscription to a data product, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data product that has a subscription request. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.

4. Choose the **Inventory** tab, and then choose the **Data products** filter. This displays existing data products in the project inventory.
5. Choose the data product that has a subscription request.
6. Choose the **Subscription requests** tab.
7. Locate the request that you want to review and then choose **View request**.
8. In the **Subscription request** window, type in a decision comment. Then choose either **Approve** or **Reject**.

Republish data products in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables data producers to group data assets into well-defined, self-contained packages called data products that are tailored for specific business use cases. For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#).

Any Amazon SageMaker Unified Studio user with the required permissions can republish an Amazon SageMaker Unified Studio data product.

To republish a data product complete the following steps.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data product that you want to edit. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Assets**.
4. Choose the **Inventory** tab, and then choose the **Data products** filter. This displays existing data products in the project inventory.
5. Choose the data product that you want to republish.
6. Make the desired edits to the data product. For more information, see [the section called “Edit data products”](#).
7. On the data product's details page, choose **Re-publish**. Confirm this action by choosing **Re-publish data product** in the **Re-publish data product** pop-up window.

 **Note**

Republishing this data product will update the following for all subscribers:

- If assets have been removed from the data product, subscribers will no longer have access to these assets.
- If assets have been added to the data product, subscribers will get access to these assets.
- New published versions of data assets will be available.

Data inventory and publishing

This section describes the tasks and procedures to create an inventory of your data in Amazon SageMaker Unified Studio and to publish your data in Amazon SageMaker Unified Studio.

To use Amazon SageMaker Unified Studio to catalog your data, you must first bring your data (assets) as inventory of your project in Amazon SageMaker Unified Studio. Creating an inventory for a particular project makes the assets discoverable only to that project's members. Project inventory assets are not available to all domain users in search or browse unless it is published to the Amazon SageMaker catalog. After creating a project inventory, data owners can curate their inventory assets with the required business metadata by adding or updating business names (asset and schema), descriptions (asset and schema), README, glossary terms (asset and schema), and metadata forms.

The next step of using Amazon SageMaker Unified Studio to catalog your data is to make your project's inventory assets discoverable by the domain users. You can do this by publishing the inventory assets to the Amazon SageMaker Unified Studio catalog. Only the latest version of the inventory asset can be published to the catalog and only the latest published version is active in the discovery catalog. If an inventory asset is updated after it's been published into the Amazon SageMaker Unified Studio catalog, you must publish it again for the latest version to be in the discovery catalog.

For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#)

Topics

- [Configure Lake Formation permissions for Amazon SageMaker Unified Studio](#)
- [Create custom asset types in Amazon SageMaker Unified Studio](#)
- [Create an Amazon SageMaker Unified Studio data source for AWS Glue in the project catalog](#)

- [Create an Amazon SageMaker Unified Studio data source for Amazon Redshift in the project catalog](#)
- [Edit a data source in Amazon SageMaker Unified Studio](#)
- [Delete a data source in Amazon SageMaker Unified Studio](#)
- [Publish assets to the Amazon SageMaker Unified Studio catalog from the project inventory](#)
- [Manage inventory and curate assets in Amazon SageMaker Unified Studio](#)
- [Manually create an asset in Amazon SageMaker Unified Studio](#)
- [Unpublish an asset from the Amazon SageMaker catalog](#)
- [Delete an Amazon SageMaker Unified Studio asset](#)
- [Manually start a data source run in Amazon SageMaker Unified Studio](#)
- [Asset revisions in Amazon SageMaker Unified Studio](#)
- [Data quality in Amazon SageMaker Unified Studio](#)
- [Data lineage in Amazon SageMaker Unified Studio](#)
- [Analyze Amazon SageMaker Unified Studio data with external analytics applications via JDBC connection](#)
- [Metadata enforcement rules for publishing](#)

Configure Lake Formation permissions for Amazon SageMaker Unified Studio

When you create a project in Amazon SageMaker Unified Studio, an AWS Glue database is added as part of this project. If you want to publish assets from this AWS Glue database, no additional permissions are needed.

However, if you want to publish assets and subscribe to assets from an AWS Glue database that exists outside of your Amazon SageMaker Unified Studio project, you must explicitly provide your project with the permissions to access tables in the external AWS Glue database. To do this, you must complete the following settings in AWS Lake Formation and attach necessary AWS Lake Formation permissions to the project's IAM role role.

- Configure the Amazon S3 location for your data lake in AWS Lake Formation with **Lake Formation** permission mode. For more information, see <https://docs.aws.amazon.com/lake-formation/latest/dg/register-data-lake.html>.
- Attach the following AWS Lake Formation permissions to the AWS Glue manage access role:

- **Describe** and **Describe grantable** permissions on the database where the tables exist.
- **Describe, Select, Describe Grantable, Select Grantable** permissions on all the tables in the above database that you want DataZone to manage access on your behalf.

Create custom asset types in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, assets represent specific types of data resources such as database tables, dashboards, or machine learning models. To provide consistency and standardization when describing catalog assets, an Amazon SageMaker Unified Studio domain must have a set of asset types that define how assets are represented in the catalog. An asset type defines the schema for a specific type of asset. An asset type has a set of required and optional nameable metadata form types. Asset types in Amazon SageMaker Unified Studio are versioned. When assets are created, they are validated against the schema defined by their asset type (typically latest version), and if an invalid structure is specified, asset creation fails.

System asset types - Amazon SageMaker Unified Studio provisions service-owned system asset types. System asset types cannot be altered. Amazon SageMaker Unified Studio includes the following system asset types:

- Amazon Bedrock chat app
- Amazon Bedrock flow app
- Amazon Bedrock inference only
- Amazon Bedrock model
- Amazon Bedrock prompt
- Databricks table
- Databricks view
- AWS Glue table
- AWS Glue view
- Amazon Redshift table
- Amazon Redshift view
- Amazon S3 object collection
- SageMaker feature group
- SageMaker model package group
- Snowflake table

- Snowflake view
- Data product

Custom asset types - to create custom asset types, you start by creating the required metadata form types and glossaries to use in the form types. You can then create custom asset types by specifying a name, description, and associated metadata forms that can be required or optional.

For asset types with structured data, to represent the column schema in Amazon SageMaker Unified Studio, you can use the `RelationalTableFormType` to add the technical metadata to your columns, including column names, descriptions, and data types, and the `ColumnBusinessMetadataForm` to add the business descriptions of the columns, including business names, glossary terms, and custom key value pairs.

To create a custom asset type in Amazon SageMaker Unified Studio, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project where you want to create a custom asset type.
3. Navigate to the **Discover** menu in the top navigation.
4. Choose **Data catalog**.
5. Choose **View asset types**.
6. Choose **Create asset type**.
7. Specify the following:
 - **Name** - the name of the custom asset type
 - **Description** - the description of the custom asset type.
 - Choose **Add metadata form** to add metadata forms to this custom asset type.
 - Under **Usage permission**, restrict access, by specify which projects or domain units are authorized to use this asset type.

 **Note**

You must be a domain unit owner or a project owner in order to modify usage permissions. Project contributors can view usage permissions but cannot edit them.

You can choose the following:

- **All projects** - give permissions to all projects in this domain
- **Owning project** - give permissions only to the owning project
- **Selected projects or domain units** - give permissions to specific projects and/or domain units

If you select this option, choose **Add usage permission**, and in the **Add projects and designations** pop up window, specify the authorized projects (you can choose **Select projects in a domain unit** or **All project in a domain unit**), the specific domain unit, and the allowed designations - which designations a project member must have to use this policy. You can choose **Owner** or **Contributor**.

8. Choose **Create**. After the custom asset type is created, you can use it to create assets.

Create an Amazon SageMaker Unified Studio data source for AWS Glue in the project catalog

In Amazon SageMaker Unified Studio, you can create an AWS Glue Data Catalog data source in order to import technical metadata of database tables from AWS Glue. To add a data source for the AWS Glue Data Catalog, the source database must already exist in AWS Glue. Your Amazon SageMaker Unified Studio project's IAM role also needs certain permissions to be able to create a data source, as described in the section [the section called “Configure Lake Formation permissions for Amazon SageMaker Unified Studio”](#).

When you create and run an AWS Glue data source, you add assets from the source AWS Glue database to your Amazon SageMaker Unified Studio project's inventory. You can run your AWS Glue data sources on a set schedule or on demand to create or update your assets' technical metadata. During the data source runs, you can optionally choose to publish your assets to the Amazon SageMaker Unified Studio catalog and thus make them discoverable by all domain users. You can also publish your project inventory assets after editing their business metadata. Domain users can search for and discover your published assets, and request subscriptions to these assets.

Note

Adding a data source in the project catalog makes it possible to publish that data into the Amazon SageMaker catalog. To add a data source for analyzing and editing within your

project, use the **Data** page of your project. Data that you add to your connect to on the **Data** page can also be published to the Amazon SageMaker catalog. For more information, see [Amazon SageMaker Lakehouse](#).

To create an AWS Glue data source

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which you want to add the data source.
3. Choose **Data sources** from the left navigation pane under **Project catalog**.
4. Choose **Create data source**.
5. Configure the following fields:
 - **Name** – The data source name.
 - **Description** – The data source description.
6. Under **Data source type**, choose **AWS Glue**.
7. (Optional) Under **Connection**, select **Import data lineage** if you want to import lineage for the data sources that use the connection.
8. Under **Data selection**, provide an AWS Glue database and provide a catalog, database names, and criteria for tables. For example, if you choose **Include** and enter *corporate, the database will include all source tables that end with the word corporate.

You can either choose an AWS Glue catalog from the dropdown or type a catalog name. The dropdown includes the default AWS Glue catalog for the connection account.

You can add multiple include and exclude rules for tables. You can also add multiple databases using the **Add another database** button.

9. Choose **Next**.
10. For **Publishing settings**, choose whether assets are immediately discoverable in the Amazon SageMaker catalog. If you only add them to the inventory, you can choose subscription terms later and then publish them to the Amazon SageMaker catalog.
11. For **Metadata generation methods**, choose whether to automatically generate metadata for assets as they're imported from the source.

12. Under **Data quality**, you can choose to **Enable data quality for this data source**. If you do this, Amazon SageMaker Unified Studio imports your existing AWS Glue data quality output into your Amazon SageMaker Unified Studio catalog. By default, Amazon SageMaker Unified Studio imports the latest existing 100 quality reports with no expiration date from AWS Glue.

Data quality metrics in Amazon SageMaker Unified Studio help you understand the completeness and accuracy of your data sources. Amazon SageMaker Unified Studio pulls these data quality metrics from AWS Glue in order to provide context during a point in time, for example, during a business data catalog search. Data users can see how data quality metrics change over time for their subscribed assets. Data producers can ingest AWS Glue data quality scores on a schedule. The Amazon SageMaker Unified Studio business data catalog can also display data quality metrics from third-party systems through data quality APIs.

13. (Optional) For **Metadata forms**, add forms to define the metadata that is collected and saved when the assets are imported into Amazon SageMaker Unified Studio. For more information, see [the section called “Create a metadata form”](#).
14. Choose **Next**.
15. For **Run preference**, choose when to run the data source.
 - **Run on a schedule** – Specify the dates and time to run the data source.
 - **Run on demand** – You can manually initiate data source runs.
16. Choose **Next**.
17. Review your data source configuration and choose **Create**.

Create an Amazon SageMaker Unified Studio data source for Amazon Redshift in the project catalog

In Amazon SageMaker Unified Studio, you can create an Amazon Redshift data source in order to import technical metadata of database tables and views from the Amazon Redshift data warehouse. To add a Amazon SageMaker Unified Studio data source for Amazon Redshift, the source data warehouse must already exist in the Amazon Redshift.

When you create and run an Amazon Redshift data source, you add assets from the source Amazon Redshift data warehouse to your Amazon SageMaker Unified Studio project's inventory. You can run your Amazon Redshift data sources on a set schedule or on demand to create or update your assets' technical metadata. During the data source runs, you can optionally choose to publish your project inventory assets to the Amazon SageMaker Unified Studio catalog and thus make them

discoverable by all domain users. You can also publish your inventory assets after editing their business metadata. Domain users can search for and discover your published assets and request subscriptions to these assets.

Note

Adding a data source in the project catalog makes it possible to publish that data into the Amazon SageMaker catalog. To add a data source for analyzing and editing within your project, use the **Data** page of your project. Data that you add to your connect to on the **Data** page can also be published to the Amazon SageMaker catalog. For more information, see [Amazon SageMaker Lakehouse](#).

To add an Amazon Redshift data source

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which you want to add the data source.
3. Choose **Data sources** from the left navigation pane under **Project catalog**.
4. Choose **Create data source**.
5. Configure the following fields:
 - **Name** – The data source name.
 - **Description** – The data source description.
6. Under **Data source type**, choose **Amazon Redshift**.
7. Under **Connection**, select a connection for your data source. The connection cannot be changed after the data source is created.
8. Under **Data selection**, provide an Amazon Redshift database schema name and enter your table or view selection criteria. For example, if you choose **Include** and enter *corporate, the asset will include all source tables that end with the word corporate.

You can add multiple include rules. You can also add another schema using the **Add another schema** button.
9. Choose **Next**.

10. For **Publishing settings**, choose whether assets are immediately discoverable in Amazon SageMaker catalog. If you only add them to the inventory, you can choose subscription terms later and then publish them to the Amazon SageMaker catalog.
11. For **Metadata generation methods**, choose whether to automatically generate metadata for assets as they're published and updated from the source.
12. (Optional) For **Metadata forms**, add forms to define the metadata that is collected and saved when the assets are imported into Amazon SageMaker Unified Studio. For more information, see [the section called "Create a metadata form"](#).
13. Choose **Next**.
14. For **Run preference**, choose when to run the data source.
 - **Run on a schedule** – Specify the dates and time to run the data source.
 - **Run on demand** – You can manually initiate data source runs.
15. Choose **Next**.
16. Review your data source configuration and choose **Create**.

Edit a data source in Amazon SageMaker Unified Studio

After you create an Amazon SageMaker Unified Studio data source, you can modify it to change the source details or the data selection criteria. When you no longer need a data source, you can delete it.

To edit a data source in the project catalog

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project that contains the data source that you want to edit.
3. Choose **Data sources** from the left navigation pane under **Project catalog**.
4. Choose the data source that you want to modify.
5. Expand the **Actions** menu, then choose **Edit data source**.
6. Make your changes to the data source fields as desired, then choose **Save**.

Delete a data source in Amazon SageMaker Unified Studio

When you no longer need an Amazon DataZone data source, you can remove it permanently. After you delete a data source, all assets that originated from that data source are still available in the catalog, and users can still subscribe to them. However, the assets will stop receiving updates from the source. We recommend that you first move the dependent assets to a different data source before you delete it.

To delete a data source in the project catalog

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project that contains the data source that you want to edit.
3. Choose **Data sources** from the left navigation pane under **Project catalog**.
4. Choose the data source that you want to delete.
5. Expand the **Actions** menu, then choose **Delete data source**.
6. To confirm deletion, type delete in the text entry field. Then choose **Delete**.

Publish assets to the Amazon SageMaker Unified Studio catalog from the project inventory

You can publish Amazon SageMaker Unified Studio assets and their metadata from project inventories into the Amazon SageMaker Unified Studio catalog. You can only publish the most recent version of an asset to the catalog.

Consider the following when publishing assets to the catalog:

- To publish an asset to the catalog, you must be the owner or contributor of the project that contains the asset.
- For Amazon Redshift assets, ensure that the Amazon Redshift clusters associated with both publisher and subscriber clusters meet all the requirements for Amazon Redshift data sharing in order for Amazon SageMaker Unified Studio to manage access for Redshift tables and views. See [Data sharing concepts for Amazon Redshift](#).

Publish an asset in Amazon SageMaker Unified Studio

If you didn't choose to make assets immediately discoverable in the data catalog when you created a data source, perform the following steps to publish them later.

To publish an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. Make sure you are on the **Inventory** tab, then choose the name of the asset that you want to publish. You are then brought to the asset details page.

 **Note**

By default, all assets require subscription approval, which means a data owner must approve all subscription requests to the asset. If you want to change this setting before publishing the asset, open the asset details and choose **Edit** next to **Subscription approval**. You can change this setting later by modifying and re-publishing the asset.

5. Choose **Publish asset**. The asset is directly published to the catalog.

If you make changes to the asset, such as modifying its approval requirements, you can choose **Re-publish asset** to publish the updates to the catalog.

Manage inventory and curate assets in Amazon SageMaker Unified Studio

In order to use Amazon SageMaker Unified Studio to catalog your data, you must first bring your data (assets) as inventory of your project in Amazon SageMaker Unified Studio. Creating inventory for a particular project makes the assets discoverable only to that project's members.

After the assets are created in project inventory, their metadata can be curated. For example, you can edit the asset's name, description, or README. Each edit to the asset creates a new version of the asset. You can use the History tab on the asset's details page to view all asset versions.

You can edit the **README** section and add rich descriptions for the asset. The **README** section supports markdown, thus enabling you to format your descriptions as required and describe key information about an asset to consumers.

Glossary terms can be added at the asset level by filling out available forms.

To curate the schema, you can review the columns, add business names, descriptions, and add glossary terms at column level.

If automated metadata generation is enabled when the data source is created, the business names for assets and columns are available to review and accept or reject individually or all at once.

You can also edit the subscription terms to specify if approval for the asset is required or not.

Metadata forms in Amazon SageMaker Unified Studio enable you to extend a data asset's metadata model by adding custom-defined attributes (for example, sales region, sales year, and sales quarter). The metadata forms that are attached to an asset type are applied to all assets created from that asset type. You can also add additional metadata forms to individual assets as part of the data source run or after it's created. For creating new forms, see [the section called "Create a metadata form".](#)

To update the metadata of an asset, you must be the owner or the contributor of the project to which the asset belongs.

To update the metadata of an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. Make sure you are on the **Inventory** tab, then choose the name of the asset that you want to publish. You are then brought to the asset details page.
5. On the asset details page, under **Metadata forms**, choose **Edit values** to edit the existing forms as needed, or choose **Add metadata form** and enter values for each of the metadata fields to attach additional metadata forms to the asset.
6. When you're done making updates, choose **Save**.

When you save the form, Amazon SageMaker Unified Studio generates a new inventory version of the asset. To publish the updated version to the catalog, choose **Re-publish asset**.

By default, metadata forms attached to a domain are attached to all assets published to that domain. Data publishers can associate additional metadata forms to individual assets in order to provide additional context.

When you are satisfied with the asset curation, the data owner can publish an asset version to the Amazon SageMaker Unified Studio catalog and thus make it discoverable by all domain users. The asset in the project shows the inventory version and the published version. In the discovery catalog, only the latest published version appears. If the metadata is updated after publishing, then a new inventory version will be available for publishing to the catalog.

Manually create an asset in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, an asset is an entity that presents a single physical data object (for example, a table, a dashboard, a file) or virtual data object (for example, a view). For more information, see [Amazon SageMaker Unified Studio terminology and concepts](#). Publishing an asset manually is a one-time operation. You don't specify a run schedule for the asset, so it's not updated automatically if its source changes.

To manually create an asset through a project, you must be the owner or contributor of that project.

To create an asset manually

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project that you want to create an asset in.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. On the **Inventory** tab, choose **Create**, then choose **Create asset**. choose **Create asset**.
5. For **Data asset details**, configure the following settings:
 - **Name** – The name of the asset.
 - **Description** – A description of the asset.
6. Choose **Next**.

7. For **Asset type details**, configure the following settings:
 - **Asset type** – The type of asset.
 - **Revision**.
8. If you are adding an **S3 object collection**, for **S3 location**, enter the Amazon Resource Name (ARN) of the source S3 bucket.

Optionally, enter an S3 access point. For more information, see [Managing data access with Amazon S3 access points](#).
9. Choose **Next**.
10. Review the selections, then choose **Create**.

After the asset is created, it will be stored in the inventory until you decide to publish it.

Unpublish an asset from the Amazon SageMaker catalog

When you unpublish an Amazon SageMaker Unified Studio asset from the catalog, it no longer appears in global search results. New users won't be able to find or subscribe to the asset listing in the catalog, but all existing subscriptions remain the same.

To unpublish an asset, you must be the owner or the contributor of the project to which the asset belongs.

To unpublish an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. On the **Inventory** tab, choose the name of the asset that you want to unpublish. This opens the asset details page.
5. Expand the **Actions** menu, then choose **Unpublish**.
6. In the pop-up window, confirm the action by choosing **Unpublish**.

The asset is then removed from the catalog. You can re-publish the asset at any time by choosing **Publish asset**.

Delete an Amazon SageMaker Unified Studio asset

When you no longer need an asset in Amazon SageMaker Unified Studio, you can permanently delete it. Deleting an asset is different than unpublishing an asset from the catalog. You can delete an asset and its related listing in the catalog so that it's not visible in any search results. To delete the asset listing, you must first revoke all of its subscriptions.

To delete an asset, you must be the owner or the contributor of the project to which the asset belongs.

 **Note**

In order to delete an asset listing, you must first revoke all existing subscriptions to the asset, and the asset must be removed from all data products. You can't delete an asset listing that has existing subscribers or that is included in a current data product.

To delete an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. On the **Inventory** tab, choose the name of the asset that you want to unpublish. This opens the asset details page.
5. Expand the **Actions** menu, then choose **Delete**.
6. In the pop-up window, type delete to confirm deletion, then choose **Delete**.

When the asset is deleted, it's no longer available to view or subscribe to.

Manually start a data source run in Amazon SageMaker Unified Studio

When you run a data source, Amazon SageMaker Unified Studio pulls all any new or modified metadata from the source and updates the associated assets in the inventory. When you add a data source to Amazon SageMaker Unified Studio, you specify the source's run preference, which defines

whether the source runs on a schedule or on demand. If your source runs on demand, you must initiate a data source run manually.

Even if your source runs on a schedule, you can still run it manually at any time. After adding business metadata to the assets, you can select assets and publish them to the Amazon SageMaker catalog in order for these assets to be discoverable by all domain users. Only published assets are searchable by other domain users.

To run a data source manually

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the data source belongs.
3. Choose **Data sources** from the left navigation pane under **Project catalog**.
4. Choose the data source that you want to run. This opens the data source details page.
5. Choose **Run**.

The data source status changes as Amazon SageMaker Unified Studio updates the asset metadata with the most recent data from the source. You can monitor the status of the run on the **Data source runs** tab.

Asset revisions in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio increments the revision of an asset when you edit its business or technical metadata. These edits include modifying the asset name, description, glossary terms, column names, metadata forms, and metadata form field values. These changes can result from manual edits, data source job runs, or API operations. Amazon SageMaker Unified Studio automatically generates a new asset revision any time you make an edit to the asset.

After you update an asset and a new revision is generated, you must publish the new revision to the catalog for it to be updated and available to subscribers. For more information, see [the section called "Publish assets to the catalog from the project inventory"](#). You can only publish the most recent version of an asset to the catalog.

To view past revisions of an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. On the **Inventory** tab, choose the name of the asset that you want to unpublish. This opens the asset details page.
5. Navigate to the **History** tab, which displays a list of past revisions of the asset.

Data quality in Amazon SageMaker Unified Studio

Data quality metrics in Amazon SageMaker Unified Studio help you understand the different quality metrics such as completeness, timeliness, and accuracy of your data sources. Amazon SageMaker Unified Studio integrates with AWS Glue Data Quality and offers APIs to integrate data quality metrics from third-party data quality solutions. Data users can see how data quality metrics change over time for their subscribed assets. To author and run the data quality rules, you can use your data quality tool of choice such as AWS Glue data quality. With data quality metrics in Amazon DataZone, data consumers can visualize the data quality scores for the assets and columns, helping build trust in the data they use for decisions.

Prerequisites and IAM role changes

If you are using Amazon SageMaker Unified Studio's AWS managed policies, there are no additional configuration steps and these managed policies are automatically updated to support data quality. If you are using your own policies for the roles that grant Amazon SageMaker Unified Studio the required permissions to interoperate with supported services, you must update the policies attached to these roles to enable support for reading the AWS Glue data quality information.

Enabling data quality for AWS Glue assets

Amazon SageMaker Unified Studio pulls the data quality metrics from AWS Glue in order to provide context during a point in time, for example, during a business data catalog search. Data users can see how data quality metrics change over time for their subscribed assets. Data producers can ingest AWS Glue data quality scores on a schedule. The Amazon SageMaker Unified Studio business data catalog can also display data quality metrics from third-party systems through data

quality APIs. For more information, see [AWS Glue Data Quality](#) and [Getting started with AWS Glue Data Quality for the Data Catalog](#).

You can enable data quality metrics for your Amazon SageMaker Unified Studio assets in the following ways:

- Use Amazon SageMaker Unified Studio or the Amazon DataZone APIs to enable data quality for your AWS Glue data source via the Amazon SageMaker Unified Studio either while creating new or editing existing AWS Glue data source.

 **Note**

You can use Amazon SageMaker Unified Studio to enable data quality only for your AWS Glue inventory assets. In this release of Amazon SageMaker Unified Studio, enabling data quality for custom types assets in Amazon SageMaker Unified Studio must be done using APIs.

- You can also use the APIs to enable data quality for your new or existing data sources. You can do this by invoking the [CreateDataSource](#) or [UpdateDataSource](#) APIs and setting the `autoImportDataQualityResult` parameter to 'True'.

After data quality is enabled, you can run the data source on demand or on schedule. Each run can bring in up to 100 metrics per asset. There is no need to create forms or add metrics manually when using data source for data quality. When the asset is published, the updates that were made to the data quality form (up to 30 data points per rule of history) are reflected in the listing for the consumers. Subsequently, each new addition of metrics to the asset is automatically added to the listing. There is no need to republish the asset to make the latest scores available to consumers.

Enabling data quality for custom asset types

You can use the Amazon SageMaker Unified Studio APIs to enable data quality for any of your custom type assets. For more information, see the following:

- [PostTimeSeriesDataPoints](#)
- [ListTimeSeriesDataPoints](#)
- [GetTimeSeriesDataPoint](#)
- [DeleteTimeSeriesDataPoints](#)

The following steps provide an example of using APIs or CLI to import third-party metrics for your assets in Amazon SageMaker Unified Studio:

1. Invoke the `PostTimeSeriesDataPoints` API as follows:

```
aws datazone post-time-series-data-points \
--cli-input-json file://createTimeSeriesPayload.json \
```

with the following payload:

```
"domainId": "dzd_5oo7xzoqltu8mf",
  "entityId": "4wyh64k2n8czaf",
  "entityType": "ASSET",
  "form": {
    "content": "{\n      \"evaluations\": [ {\n        \"types\": [ \"MaximumLength\" ],\n        \"description\": \"ColumnLength \\\"ShippingCountry\\\" <= 6\", \"details\": { },\n        \"applicableFields\": [ \"ShippingCountry\" ],\n        \"status\": \"PASS\"\n      }, {\n        \"types\": [ \"MaximumLength\" ],\n        \"description\": \"ColumnLength \\\"ShippingState\\\" <= 2\", \"details\": { },\n        \"applicableFields\": [ \"ShippingState\" ],\n        \"status\": \"PASS\"\n      }, {\n        \"types\": [ \"MaximumLength\" ],\n        \"description\": \"ColumnLength \\\"ShippingCity\\\" <= 8\", \"details\": { },\n        \"applicableFields\": [ \"ShippingCity\" ],\n        \"status\": \"PASS\"\n      }, {\n        \"types\": [ \"Completeness\" ],\n        \"description\": \"Completeness \\\"ShippingStreet\\\" >= 0.59\", \"details\": { },\n        \"applicableFields\": [ \"ShippingStreet\" ],\n        \"status\": \"PASS\"\n      }, {\n        \"types\": [ \"MaximumLength\" ],\n        \"description\": \"ColumnLength \\\"BillingCountry\\\" <= 6\", \"details\": { },\n        \"applicableFields\": [ \"BillingCountry\" ],\n        \"status\": \"PASS\"\n      }, {\n        \"types\": [ \"Completeness\" ],\n        \"description\": \"Completeness \\\"Billingcountry\\\" >= 0.5\", \"details\": { },\n        \"EVALUATION_MESSAGE\": \"Value: 0.2666666666666666 does not meet the constraint requirement!\"\n      }, {\n        \"types\": [ \"Completeness\" ],\n        \"description\": \"Completeness \\\"Billingstreet\\\" >= 0.5\", \"details\": { },\n        \"applicableFields\": [ \"Billingstreet\" ],\n        \"status\": \"FAIL\"\n      }, {\n        \"types\": [ \"Completeness\" ],\n        \"description\": \"Completeness \\\"Billingstreet\\\" >= 0.5\", \"details\": { },\n        \"applicableFields\": [ \"Billingstreet\" ],\n        \"status\": \"PASS\"\n      } ],\n      \"passingPercentage\": 88.0,\n      \"evaluationsCount\": 8\n    }\n  }",\n  "status": "PENDING",\n  "lastModified": "2023-08-15T12:00:00Z",\n  "version": 1\n}
```

```
        "formName": "shortschemaruleset",
        "id": "athp9dyw75gzhj",
        "timestamp": 1.71700477757E9,
        "typeIdentifier": "amazon.datazone.DataQualityResultFormType",
        "typeRevision": "8"
    },
    "formName": "shortschemaruleset"
}
```

You can obtain this payload by invoking the GetFormType action:

```
aws datazone get-form-type --domain-identifier <your_domain_id> --form-type-  
identifier amazon.datazone.DataQualityResultFormType --region <domain_region> --  
output text --query 'model.smithy'
```

2. Invoke the DeleteTimeSeriesDataPoints API as follows:

```
aws datazone delete-time-series-data-points\  
--domain-identifier dzd_bqqlk3nz21zp2f \  
--entity-identifier dzd_bqqlk3nz21zp2f \  
--entity-type ASSET \  
--form-name rulesET1 \  

```

Data lineage in Amazon SageMaker Unified Studio

Data lineage in Amazon SageMaker Unified Studio is an OpenLineage-compatible feature that can help you to capture and visualize lineage events, from OpenLineage-enabled systems or through APIs, to trace data origins, track transformations, and view cross-organizational data consumption. It provides you with an overarching view into your data assets to see the origin of assets and their chain of connections. The lineage data includes information on the activities inside the Amazon SageMaker catalog, including information about the catalogued assets, the subscribers of those assets, and the activities that happen outside the business data catalog captured programmatically using the APIs.

Using Amazon SageMaker Unified Studio's OpenLineage-compatible APIs, domain administrators and data producers can capture and store lineage events beyond what is available in Amazon

SageMaker Unified Studio, including transformations in Amazon S3, AWS Glue, and other services. This provides a comprehensive view for the data consumers and helps them gain confidence of the asset's origin, while data producers can assess the impact of changes to an asset by understanding its usage. Additionally, Amazon SageMaker Unified Studio versions lineage with each event, enabling users to visualize lineage at any point in time or compare transformations across an asset's or job's history. This historical lineage provides a deeper understanding of how data has evolved, essential for troubleshooting, auditing, and ensuring the integrity of data assets.

With data lineage, you can accomplish the following in Amazon SageMaker Unified Studio:

- Understand the provenance of data: knowing where the data originated fosters trust in data by providing you with a clear understanding of its origins, dependencies, and transformations. This transparency helps in making confident data-driven decisions.
- Understand the impact of changes to data pipelines: when changes are made to data pipelines, lineage can be used to identify all of the downstream consumers that are to be affected. This helps to ensure that changes are made without disrupting critical data flows.
- Identify the root cause of data quality issues: if a data quality issue is detected in a downstream report, lineage, especially column-level lineage, can be used to trace the data back (at a column level) to identify the issue back to its source. This can help data engineers to identify and fix the problem.
- Improve data governance and compliance: column-level lineage can be used to demonstrate compliance with data governance and privacy regulations. For example, column-level lineage can be used to show where sensitive data (such as PII) is stored and how it is processed in downstream activities.

Types of lineage nodes in Amazon SageMaker Unified Studio

In Amazon SageMaker Unified Studio, data lineage information is presented in nodes that represent tables and views. Depending on the context of the project, the producers are able to view both inventory and published assets, whereas consumers can only view the published assets. When you first open the lineage tab in the asset details page, the catalogued dataset node is the starting point for navigating upstream or downstream through the lineage nodes of your lineage graph.

The following are the types of data lineage nodes that are supported in Amazon SageMaker Unified Studio:

- **Dataset node** - this node type includes data lineage information about a specific data asset.

- Dataset nodes that include information about AWS Glue or Amazon Redshift assets published in the Amazon SageMaker Unified Studio catalog are auto-generated and include a corresponding AWS Glue or Amazon Redshift icon within the node.
- Dataset nodes that include information about assets that are not published in the Amazon SageMaker Unified Studio catalog, are created manually by domain administrators (producers) and are represented by a default custom asset icon within the node.
- **Job (run) node** - this node type displays the details of the job, including the latest run of a particular job and run details. This node also captures multiple runs of the job and can be viewed in the **History** tab of the node details. You can view node details by choosing the node icon.

Key attributes in lineage nodes

The `sourceIdentifier` attribute in a lineage node represents the events happening on a dataset. The `sourceIdentifier` of the lineage node is the identifier of the dataset (table/view etc). It's used for uniqueness enforcement on the lineage nodes. For example, there can't be two lineage nodes with same `sourceIdentifier`. The following are examples of `sourceIdentifier` values for different types of nodes:

- For dataset node with respective dataset type:
 - Asset: `amazon.datazone.asset/<assetId>`
 - Listing (published asset): `amazon.datazone.listing/<listingId>`
 - AWS Glue table: `arn:aws:glue:<region>:<account-id>:table/<database>/<table-name>`
 - Amazon Redshift table/view: `arn:aws:<redshift/redshift-serverless>:<region>:<account-id>:<table-type(table/view etc)>/<clusterIdentifier/workgroupName>/<database>/<schema>/<table-name>`
 - For any other type of dataset nodes imported using open-lineage run events, `<namespace>/<name>` of the input/output dataset is used as `sourceIdentifier` of the node.
- For jobs:
 - For job nodes imported using open-lineage run events, `<jobs_namespace>. <job_name>` is used as `sourcelIdentifier`.
- For job runs:
 - For job run nodes imported using open-lineage run events, `<jobs_namespace>. <job_name>/<run_id>` is used as `sourcelIdentifier`.

For assets created using `createAsset` API, the `sourceIdentifier` must be updated using `createAssetRevision` API to enable mapping the asset to upstream resources.

Visualizing data lineage

Amazon SageMaker Unified Studio's asset details page provides a graphical representation of data lineage, making it easier to visualize data relationships upstream or downstream. The asset details page provides the following capabilities to navigate the graph:

- Column-level lineage: expand column-level lineage when available in dataset nodes. This automatically shows relationships with upstream or downstream dataset nodes if source column information is available.
- Column search: when the default display for number of columns is 10. If there are more than 10 columns, pagination is activated to navigate to the rest of the columns. To quickly view a particular column, you can search on the dataset node that list just the searched column.
- View dataset nodes only: if you want to toggle to view only dataset lineage nodes and filter out the job nodes, you can choose the Open view control icon on the top left of the graph viewer and toggle the **Display dataset nodes only** option. This will remove all the job nodes from the graph and lets you navigate just the dataset nodes. Note that when the view only dataset nodes is turned on, the graph cannot be expanded upstream or downstream.
- Details pane: Each lineage node has details captured and displayed when selected.
 - Dataset node has a detail pane to display all the details captured for that node for a given timestamp. Every dataset node has 3 tabs, namely: Lineage info, Schema, and History tab. The history tab lists the different versions of lineage event captured for that node. All details captured from API are displayed using metadata forms or a JSON viewer.
 - Job node has a detail pane to display job details with tabs, namely: Job info, and History. The details pane also captures query or expressions captured as part of the job run. The history tab lists the different versions of job run event captured for that job. All details captured from API are displayed using metadata forms or a JSON viewer.
 - Version tabs: all lineage nodes in Amazon SageMaker Unified Studio data lineage have versioning. For every dataset node or job node, the versions are captured as history and that enables you to navigate between the different versions to identify what has changed overtime. Each version opens a new tab in the lineage page to help compare or contrast.

Data lineage authorization in Amazon SageMaker Unified Studio

Write permissions - to publish lineage data into Amazon SageMaker Unified Studio, you must have an IAM role with a permissions policy that includes an ALLOW action on the PostLineageEvent API. This IAM authorization happens at API Gateway layer.

Read permissions - there are two operations: GetLineageNode and ListLineageNodeHistory that are included in the AmazonDataZoneDomainExecutionRolePolicy managed policy configured by your admin. This means that every user in the Amazon SageMaker Unified Studio domain can invoke these to traverse the data lineage graph.

Data lineage sample experience in Amazon SageMaker Unified Studio

You can use the data lineage sample experience to browse and understand data lineage in Amazon SageMaker Unified Studio, including traversing upstream or downstream in your data lineage graph, exploring versions and column-level lineage.

Complete the following procedure to try the sample data lineage experience in Amazon SageMaker Unified Studio:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project you want to view lineage in.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. On the **Inventory** tab, choose the name of the asset that you want to view lineage for. This opens the asset details page.
5. On the asset details page, choose the **Lineage** tab.
6. In the data lineage window, choose the info icon that says **Try sample data lineage**. Then choose **Launch**. A new pop-up window appears.
7. Choose **Start guided data lineage tour**.
8. Select a guided tour option, and then choose **Start tour**.

At this point, a tab that provides all the space of lineage information is displayed. The sample data lineage graph is initially displayed with a base node with 1-depth at either ends, upstream and downstream. You can expand the graph upstream or downstream. The columns information is also available for you to choose and see how lineage flows through the nodes.

Automate lineage capture from Data connections

Configure automated lineage capture for AWS Glue (Lakehouse) connections

As databases and tables are added to the Amazon SageMaker Unified Studio's catalog, the lineage extraction can be automated from source for those assets using data source runs in Create Connection workflow. For every connection created, lineage is not automatically enabled.

To enable lineage capture for an AWS Glue connection

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which you want to add the data source.
3. Choose **Data sources** from the left navigation pane under Project catalog.
4. Choose the data source that you want to modify.
5. Expand the **Actions** menu, then choose **Edit data source** or click on the Data Source run name to view the details and go to **Data Source Definition** tab and choose **Edit in Connection** details.
6. Go to the connections and select **Import data lineage** checkbox to configure lineage capture from the source.
7. Make other changes to the data source fields as desired, then choose **Save**.

Limitations

The lineage collection in data source runs fetches information from table metadata to build lineage. AWS Glue crawler supports different types of sources for which lineage is captured, including Amazon S3, DynamoDB, Catalog, Delta Lake, Iceberg tables, and Hudi tables stored in Amazon S3. JDBC and DocumentDB or MongoDB are currently NOT supported as sources.

If the number of tables is more than 100, the lineage run fails after 100 tables. Make sure that the AWS Glue crawler is not configured to bring in more than 100 tables in a run.

Note

When enabled, the lineage runs asynchronously to capture metadata from the source and generate lineage events to be stored in SageMaker Catalog to be visualized from a

particular asset. The status of lineage runs for the data source can be viewed along with data source run details. The lineage run is set up to run once daily. For the first run, after enabling the feature, the first pull is scheduled for ~5 minutes after and set for a daily run. You can configure specific time programmatically.

Configure automated lineage capture for Amazon Redshift connections

Capturing lineage from Amazon Redshift can be automated when the connection is added to an Amazon Redshift source in Amazon SageMaker Unified Studio's Data explorer. Lineage capture can be automated for a connection at the data source configuration. For every connection created, lineage is not automatically enabled.

To enable lineage capture for an Amazon Redshift connection

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which you want to add the data source.
3. Choose **Data sources** from the left navigation pane under **Project catalog**.
4. Choose the data source that you want to modify.
5. Expand the **Actions** menu, then choose **Edit data source** or click on the data source run name to view the details and go to Data Source Definition tab and select **Edit in Connection details**.
6. Go to the connections and select **Import data lineage** checkbox to configure lineage capture from the source.
7. Make other changes to the data source fields as desired, then choose **Save**.

Note

When enabled, the lineage runs captures queries executed for a given database and generates lineage events to be stored in Amazon DataZone to be visualized from a particular asset. The lineage run for Amazon Redshift is set up for a daily run to pull from the Amazon Redshift system tables to derive lineage. For the first run, after enabling the feature, the first pull is scheduled for ~5 minutes after and set for a daily run. You can configure specific time programmatically.

Automate lineage capture from tools

Capture lineage for Spark executions in Visual ETL

When a new job is created in vETL in Amazon SageMaker Unified Studio, lineage is automatically enabled. When a Visual ETL flow is created, lineage capture for that ETL flow is automatically enabled when you choose **Save**.

The following Spark configuration parameters are automatically added to the job being executed:

```
{  
    "--conf": "spark.extraListeners=io.openlineage.spark.agent.OpenLineageSparkListener  
    --conf spark.openlineage.transport.type=amazon_datazone_api  
    --conf spark.openlineage.transport.domainId={DOMAIN_ID}  
    --conf spark.glue.accountId={ACCOUNT_ID}  
    --conf  
    spark.openlineage.facets.custom_environment_variables=[AWS_DEFAULT_REGION;GLUE_VERSION;GLUE_COMMAND_CRITERIA;GLUE_PYTHON_VERSION;  
    --conf spark.glue.JOB_NAME={JOB_NAME}"  
}
```

The parameters are auto-configured and do not need any updates from the user. To understand the parameters in detail:

- `spark.extraListeners=io.openlineage.spark.agent.OpenLineageSparkListener` - OpenLineageSparkListener will be created and registered with Spark's listener bus
- `spark.openlineage.transport.type=amazon_datazone_api` - This is an OpenLineage specification to tell the OpenLineage Plugin to use DataZone API Transport to emit lineage events to DataZone's PostLineageEvent API. For more information, see https://openlineage.io/docs/integrations/spark/configuration/spark_conf/
- `spark.openlineage.transport.domainId={DOMAIN_ID}` - This parameter establishes the domain to which the API transport will submit the lineage events to.
- `spark.openlineage.facets.custom_environment_variables [AWS_DEFAULT_REGION;GLUE_VERSION;GLUE_COMMAND_CRITERIA;GLUE_PYTHON_VERSION;]` - The following environment variables (AWS_DEFAULT_REGION, GLUE_VERSION, GLUE_COMMAND_CRITERIA, and GLUE_PYTHON_VERSION), which AWS Glue interactive session populates, will be added to the LineageEvent

- spark.glue.accountId=<ACCOUNT_ID> - Account Id of the Glue Data Catalog where the metadata resides. This account id is used to construct Glue ARN in lineage event.
- spark.glue.JOB_NAME - Job name of the lineage event. In vETL flow, the job name is configured automatically to be spark.glue.JOB_NAME: \${projectId}.\${pathToNotebook}

Capture lineage for Spark executions in Notebooks

Sessions in notebooks does not have a concept of a job. You can map the Spark executions to lineage events by generating a unique job name for the notebook. You can use the %%configure magic with the below parameters to enable lineage capture for Spark executions in the notebook.

```
%%configure --name project.spark -f
{
    "--conf":"spark.extraListeners=io.openlineage.spark.agent.OpenLineageSparkListener
    --conf spark.openlineage.transport.type=amazon_datazone_api --
    conf spark.openlineage.transport.domainId={DOMAIN_ID}  --conf
    spark.glue.accountId={ACCOUNT_ID} --conf
    spark.openlineage.facets.custom_environment_variables=[AWS_DEFAULT_REGION;GLUE_VERSION;GLUE_COMMAND_CRITERIA;GLUE_PYTHON_VERSION;]
    --conf spark.glue.JOB_NAME={JOB_NAME}"
}
```

The following are the parameter details:

- spark.extraListeners=io.openlineage.spark.agent.OpenLineageSparkListener - OpenLineageSparkListener will be created and registered with Spark's listener bus
- spark.openlineage.transport.type=amazon_datazone_api - This is an OpenLineage specification to tell the OpenLineage Plugin to use DataZone API Transport to emit lineage events to DataZone's PostLineageEvent API. For more information, see https://openlineage.io/docs/integrations/spark/configuration/spark_conf
- spark.openlineage.transport.domainId={DOMAIN_ID} - This parameter establishes the domain to which the API transport will submit the lineage events to.
- spark.openlineage.facets.custom_environment_variables [AWS_DEFAULT_REGION;GLUE_VERSION;GLUE_COMMAND_CRITERIA;GLUE_PYTHON_VERSION;] - The following environment variables (AWS_DEFAULT_REGION , GLUE_VERSION , GLUE_COMMAND_CRITERIA, and GLUE_PYTHON_VERSION), which Glue interactive session populates, will be added to the LineageEvent

- spark.glue.accountId=<ACCOUNT_ID> - Account Id of the Glue Data Catalog where the metadata resides. This account id is used to construct Glue ARN in lineage event.
- spark.glue.JOB_NAME - Job name of the lineage event. In vETL flow, the job name is configured automatically to be spark.glue.JOB_NAME: \${projectId}.\${pathToNotebook}

Capture lineage EMR-S Spark executions from Notebooks

EMR v7.5 and greater with Spark engine has the necessary OpenLineage libraries built in. They need to be added to the spark submit properties in order to be used especially if AWS Glue is being used as the Hive metastore. The rest of the spark submit properties are similar to those used in AWS Glue jobs. Be sure to replace the {Domain ID} with your specific Amazon DataZone or Amazon SageMaker Unified Studio domain and to replace the {Account ID} with the account id where the EMR job is run.

```
{  
    --conf spark.extraListeners=io.openlineage.spark.agent.OpenLineageSparkListener  
    --conf spark.openlineage.transport.type=amazon_datazone_api  
    --conf spark.openlineage.transport.domainId={Domain ID}  
    --conf spark.jars=/usr/share/aws/datazone-openlineage-spark/lib/  
DataZoneOpenLineageSpark-1.0.jar  
    --conf spark.glue.accountId={Account ID}"  
}
```

- Replace DATAZONE_DOMAIN_ID and ACCOUNT_ID with valid values
- Amazon DataZone VPCE is deployed to EMR-S VPC
- The JOB_NAME is the Spark application name that is automaticaly set

Limitations

- OpenLineage libraries for Spark are built into AWS Glue v5.0+ for Spark DataFrames only. Does not support Dynamic DataFrames.
- OpenLineage libraries for Spark are built into Amazon EMR v7.5+ and only for EMR-S. Lineage is not supported in in EMR on EKS and EMR on EC2.
- LineageEvent has a size limit of 300KB.

Using Amazon SageMaker Unified Studio data lineage programmatically

To use the data lineage functionality in Amazon SageMaker Unified Studio, you can invoke the following Amazon DataZone APIs:

- [GetLineageEvent](#)
- [GetLineageNode](#)
- [ListLineageEvents](#)
- [ListLineageNodeHistory](#)
- [PostLineageEvent](#)

Analyze Amazon SageMaker Unified Studio data with external analytics applications via JDBC connection

Amazon SageMaker Unified Studio enables data consumers to easily locate and subscribe to data from multiple sources within a single project and analyze this data using Amazon Athena, Amazon Redshift Query Editor, and Amazon SageMaker.

Amazon SageMaker Unified Studio also supports authentication via the Athena JDBC driver that enables users to query their subscribed Amazon SageMaker Unified Studio data using popular external SQL and analytics tools, such as SQL Workbench, DBeaver, Tableau, Domino, Power BI and many others. Users can authenticate using their corporate credentials through SSO or IAM and begin analyzing their subscribed data within their Amazon SageMaker Unified Studio projects.

Amazon SageMaker Unified Studio's support of the Athena JDBC driver provides the following benefits:

- Greater tool choice for querying and visualization - data consumers can connect to Amazon SageMaker Unified Studio using their preferred tools from a wide range of analytics tools that support a JDBC connection. This enables them to continue using the software they are familiar with without the need to learn new tools for data consumption.
- Programmatic access - a JDBC connection to access-governed data via servers or custom applications enables data consumers to perform automated and more complex data operations.

You can use your JDBC URL to connect your external analytics tools to your Amazon SageMaker Unified Studio subscribed data. To obtain your JDBC URL, perform the following procedure:

Important

In the current release, Amazon SageMaker Unified Studio supports authentication using the Amazon Athena JDBC Driver. To complete this procedure, make sure that you have downloaded and installed the latest [Athena JDBC driver](#) for your analytics application of choice.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project where you have the data that you want to analyze.
3. In the **Project overview**, choose the **JDBC connection details** tab.
4. In **JDBC connection details** choose your authentication method (**Using IDC auth** or **Using IAM auth**) and then choose the icon next to **JDBC connection URL** to copy the string or the individual parameters of the JDBC URL. You can then use it to connect to your external analytics application.

When you connect your external analytics application to Amazon DataZone using your JDBC query or parameters, you invoke the `RedeemAccessToken` API. The `RedeemAccessToken` API exchanges an Identity Center access token for the `AmazonDataZoneDomainExecutionRole` credentials, which are used to call the `GetEnvironmentCredentials` API.

For more information about the authentication mechanism that uses IAM credentials to connect to Amazon DataZone-governed data in Athena, see [DataZone IAM Credentials Provider](#). For more information about the authentication mechanism that enables connecting to Amazon DataZone-governed data in Athena using IAM Identity Center, see [DataZone Idc Credentials Provider](#).

RedeemAccessToken API Reference

Request syntax

```
POST /sso/redeem-token HTTP/1.1
Content-type: application/json

{
    "domainId": "string",
```

```
"accessToken": "string"  
}
```

Request parameters

The request uses the following parameters.

DomainId

The ID of the Amazon DataZone domain.

Pattern: ^dzd[-_][a-zA-Z0-9_-]{1,36}\$

Required: yes

accessToken

The Identity Center access token.

Type: string

Required: yes

Response syntax

```
HTTP/1.1 200  
Content-type: application/json  
  
{  
  "credentials": AwsCredentials  
}
```

Response elements

credentials

The AmazonDataZoneDomainExecutionRole credentials that are used to call the GetEnvironmentCredentials API.

Type: Array of AwsCredentials objects. This data type includes the following properties:

- `accessKeyId`: AccessKeyId
- `secretAccessKey`: SecretAccessKey
- `sessionToken`: SessionToken
- `expiration`: Timestamp

accessToken

The Identity Center access token.

Type: string

Required: yes

Errors

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

ResourceNotFoundException

The specified resource cannot be found.

HTTP Status Code: 404

ValidationException

The input fails to satisfy the constraints specified by the AWS service.

HTTP Status Code: 400

InternalServerException

The request has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

Metadata enforcement rules for publishing

The metadata enforcement rules for publishing in Amazon SageMaker Unified Studio strengthen data governance by enabling domain unit owners to establish clear metadata requirements for data producers, streamlining access requests and enhancing data governance.

The feature is supported in all the AWS commercial Regions where Amazon SageMaker Unified Studio is currently available.

Domain unit owners can complete the following procedure to configure metadata enforcement in Amazon SageMaker Unified Studio:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Govern -> Domain units** from the top navigation pane and then choose the domain or the domain unit that you want to work with.
3. Choose the **Rules** tab and then choose **Add**.
4. On the **Create required metadata form rule** page, do the following and then choose **Add rule**:
 - Specify a name for your rule.
 - Under **Action**, choose **Data asset and product publishing**.
 - Under **Required forms**, choose **Add metadata form**, choose a metadata form within the domain / domain unit that you want to add to this rule, and then choose **Add**. You can add up to 5 metadata forms per rule.
 - Under **Scope**, specify with which data entities you want to associate these forms. You can choose data products and/or data assets.
 - Under **Data asset types**, specify whether the rule applies across all asset types or limit it to selected asset types.
 - Under **Projects**, specify whether the required forms will be associated with data products and/or assets published by all projects or only selected projects in this domain unit. Also, check **Cascade rule to child domain units** if you want child domain units to inherit this requirement.

Data discovery, subscription, and consumption

In Amazon SageMaker Unified Studio, after an asset is published to a domain, subscribers can discover and request a subscription to this asset. The subscription process begins with a subscriber searching for and browsing the catalog to find an asset they want. In the Amazon SageMaker catalog, they subscribe to the asset by submitting a subscription request that includes justification and the reason for the request. The owner of the asset reviews the request. They can either approve or reject the request.

After a subscription is granted, a fulfillment process starts to facilitate access to the asset for the subscriber. There are two primary modes of asset access control and fulfillment: those for Amazon SageMaker Unified Studio managed assets and those for assets that are not managed by Amazon SageMaker Unified Studio.

- **Managed assets** – Amazon SageMaker Unified Studio can manage fulfillment and permissions for managed assets, such as AWS Glue tables and Amazon Redshift tables and views.
- **Unmanaged assets** – Amazon SageMaker Unified Studio publishes standard events related to your actions (for example, approval given to a subscription request to Amazon EventBridge). You can use these standard events to integrate with other AWS services or third-party solutions for custom integrations.

Topics

- [Search for and view assets in the Amazon SageMaker Unified Studio catalog](#)
- [Request subscription to assets in Amazon SageMaker Unified Studio](#)
- [Approve or reject a subscription request in Amazon SageMaker Unified Studio](#)
- [Revoke an existing subscription in Amazon SageMaker Unified Studio](#)
- [Cancel a subscription request in Amazon SageMaker Unified Studio](#)
- [Unsubscribe from an asset in Amazon SageMaker Unified Studio](#)
- [Grant access to managed AWS Glue Data Catalog assets in Amazon SageMaker Unified Studio](#)
- [Grant access to managed Amazon Redshift assets in Amazon SageMaker Unified Studio](#)
- [Grant access for approved subscriptions to unmanaged assets in Amazon SageMaker Unified Studio](#)
- [Metadata enforcement rules for subscription requests](#)

Search for and view assets in the Amazon SageMaker Unified Studio catalog

Amazon SageMaker Unified Studio provides a streamlined way to search for data. Any Amazon SageMaker Unified Studio user with permissions to access Amazon SageMaker Unified Studio can search for assets in the Amazon SageMaker Unified Studio catalog and view asset names and the metadata assigned to them. You can take a closer look at an asset by examining its details page.

Note

To view the actual data that an asset contains, you must first subscribe to the asset and have your subscription request approved and access granted.

To search for assets in the catalog

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Data catalog**.
4. Find the asset you want to subscribe to by browsing or typing the name of the asset into the search bar.
5. To view details about a specific asset, choose the asset to open its details page. The details page includes the following information:
 - The asset name and type.
 - A description of the asset.
 - The current published revision of the asset, the owner, whether approval is required for subscriptions, and update history.
 - A **Business metadata** tab which includes glossary terms and metadata forms.
 - A **Subscription requests** tab which includes a list of subscribers to the domain.
 - A **Lineage** tab which displays a chart of past revisions of the asset.

Request subscription to assets in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio allows you to find, access and consume the assets in the Amazon SageMaker Unified Studio catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request.

You must be a member of a project in order to request subscription to an asset within that project.

To subscribe to an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the **Discover** menu in the top navigation bar.
3. Choose **Data catalog**.
4. Find the asset you want to subscribe to by browsing or typing the name of the asset into the search bar.
5. Choose the asset to which you want to subscribe, and then choose **Subscribe**.
6. In the **Subscribe** pop-up window, provide the following information:
 - The project that you want to subscribe to the asset.
 - A short justification for your subscription request.
7. Choose **Request**.

The project will be subscribed to the asset when the publisher approves your request.

To view the status of the subscription request, locate and choose the project with which you subscribed to the asset. Choose **Subscription requests** from the project left side navigation, then choose the **Outgoing requests** tab. This page lists the assets to which the project has requested access. You can filter the list by the status of the request.

Approve or reject a subscription request in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio allows you to find, access and consume the assets in the Amazon SageMaker Unified Studio catalog. When you find an asset in the catalog that you want to access, you must *subscribe* to the asset, which creates a subscription request. An approver can then approve or reject your request.

You must be a member of the owning project (the project that published the asset) to approve or reject a subscription request.

To approve or reject a subscription request

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

2. Navigate to the project that contains the asset that has a subscription request. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Subscription requests**.
4. Choose the **Incoming requests** tab.
5. Locate the request and choose **View request**. You can filter by **Requested** to see only requests that are still open.
6. Review the subscription request and reason for access, and decide whether to approve or reject it.
7. To approve, select between the two options:
 - **Full access:** If you choose to approve the subscription with full access option, the subscriber will get access to all the rows and columns in your data asset.
 - **Approve with row and column filters:** To limit access to specific rows and columns of data, you can choose the option to approve with row and column filters. For more information, see [Fine-grained access control to data](#).
 - Select **Choose filters**, and then from the drop down select one or more available filters you want to apply to the subscription.
 - To create a new filter you can choose Create new filter option, which opens a new page to create a new row or column filter. For more information, see [Create column filters in Amazon SageMaker Unified Studio](#) and [Create row filters in Amazon SageMaker Unified Studio](#).
8. (Optional) Enter a response that explains your reason for accepting or rejecting the request.
9. Choose either **Approve** or **Reject**.

As the project owner, you can revoke the subscription at any time. For more information, see [the section called “Revoke an existing subscription”](#).

 **Note**

Amazon SageMaker Unified Studio supports fine-grained access control for AWS Glue tables, Amazon Redshift tables, and Amazon Redshift views.

Revoke an existing subscription in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio allows you to find, access and consume the assets in the Amazon SageMaker Unified Studio catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request. You might need to revoke a subscription after you have approved it, either because the approval was a mistake, or because the subscriber no longer needs access to the asset.

You must be a member of the owning project (the project that published the asset) to revoke a subscription.

To revoke a subscription

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the asset that has a subscription request. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Subscription requests**.
4. Choose the **Incoming requests** tab.
5. Locate the subscription you want to revoke and choose **View subscription**.
6. (Optional) Enable the checkbox to allow the subscriber to keep the asset in the project's subscription targets. A subscription target is a reference to a set of resources where subscribed data can be made available within an environment.

If you want to revoke access to the asset from the subscription target at a later time, you must do so in AWS Lake Formation.

7. Choose **Revoke subscription**.

You can't re-approve a subscription after you revoke it. The subscriber must request a subscription to the asset again in order for you to approve it.

Cancel a subscription request in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio allows you to find, access and consume the assets in the Amazon SageMaker Unified Studio catalog. When you find an asset in the catalog that you want

to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request. You might need to cancel a pending subscription request, either because you submitted it by mistake, or because you no longer need read access to the asset.

To cancel a subscription request, you must be either a project owner or contributor.

To cancel a subscription request

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the asset that has a subscription request. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Subscription requests**.
4. Choose the **Outgoing requests** tab.
5. Filter by **Requested** to see only requests that are still pending.
6. Locate the request and choose **View request**.
7. Review the subscription request and choose **Cancel request**.

If you want to re-subscribe to the asset (or to a different asset), see [the section called “Request subscription to assets”](#).

Unsubscribe from an asset in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio allows you to find, access and consume the assets in the Amazon SageMaker Unified Studio catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request. You might need to unsubscribe from an asset, either because you subscribed by mistake and were approved, or because you no longer need read access to the asset.

You must be a member of a project in order to unsubscribe from one of its assets.

To unsubscribe from an asset

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

2. Navigate to the project that contains the asset that has a subscription request. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. Under **Project catalog**, choose **Subscription requests**.
4. Choose the **Outgoing requests** tab.
5. Filter by **Approved** to see only requests that have been approved.
6. Locate the request and choose **View subscription**.
7. Review the subscription and choose **Unsubscribe**.

If you want to re-subscribe to the asset (or to a different asset), see [the section called “Request subscription to assets”](#).

Grant access to managed AWS Glue Data Catalog assets in Amazon SageMaker Unified Studio

Note

Access management for the AWS Glue Data Catalog assets using the AWS Lake Formation LF-TBAC method is not supported.

Support for cross-Region sharing of assets in AWS Glue Data Catalog is not supported.

Support for cross-account sharing of assets in a federated catalog within AWS Glue Data Catalog is not supported.

When a subscription request to managed AWS Glue Data Catalog assets is approved, Amazon SageMaker Unified Studio grants and manages access to the approved AWS Glue Data Catalog tables on your behalf through AWS Lake Formation. For the subscriber project, assets that are granted appear in the AWS Glue Data Catalog as resources in your account. You can then use Amazon Athena, Amazon Redshift, or Spark to query the tables.

For Amazon SageMaker Unified Studio to be able to grant access to AWS Glue Data Catalog tables, the following conditions must be met.

- The AWS Glue table must be Lake Formation-managed since Amazon SageMaker Unified Studio grants access by managing Lake Formation permissions.

- The IAM role of the project that has published the asset to the Amazon SageMaker catalog must have the following AWS Lake Formation permissions:
 - DESCRIBE and DESCRIBE GRANTABLE permissions on the AWS Glue database that contains the published table.
 - DESCRIBE, SELECT, DESCRIBE GRANTABLE, SELECT GRANTABLE permissions in Lake Formation on the published table itself.

For more information, see [Granting and revoking permissions on catalog resources](#) in the *AWS Lake Formation Developer Guide*.

Grant access to managed Amazon Redshift assets in Amazon SageMaker Unified Studio

When a subscription to an Amazon Redshift table or view is approved, Amazon SageMaker Unified Studio can automatically add the subscribed asset to the Amazon Redshift Serverless workgroup created for the project, so that members of the project can query the data using the Amazon Redshift query editor link within the project. Under the hood, Amazon SageMaker Unified Studio creates the necessary grants and datashares.

The process of granting access varies depending on where the source database (publisher) and the target database (subscriber) are located.

- Same cluster, same database - if data must be shared within the same database, Amazon SageMaker Unified Studio grants permissions directly on the source table.
- Same cluster, different database - if data must be shared across two databases within the same cluster, Amazon SageMaker Unified Studio creates a view in the target database and permissions are granted on the created view.
- Same account different cluster - Amazon SageMaker Unified Studio creates a datashare between the source and target cluster and creates a view on top of the shared table. Permissions are granted on the view.
- Cross-account - same as above but an additional step is required to authorize cross-account datashare on the producer cluster side and another step to associate the data share on consumer cluster side.

Make sure that your publishing and subscribing Amazon Redshift clusters meet all requirements for Amazon Redshift datashares. For more information, see [Data sharing in Amazon Redshift](#) in the Amazon Redshift Developer Guide.

 **Note**

Cross-Region data sharing using Amazon Redshift is not supported.

Grant access for approved subscriptions to unmanaged assets in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables users to publish any type of asset in the Amazon SageMaker catalog. For some of these assets, Amazon SageMaker Unified Studio can automatically manage access grants. These assets are called **managed assets** and include Lake Formation-managed AWS Glue Data Catalog tables and Amazon Redshift tables and views. All other assets to which Amazon SageMaker Unified Studio can't automatically grant subscriptions are called **unmanaged**.

Amazon SageMaker Unified Studio provides a path for you to manage access grants for your unmanaged assets. When a subscription to an asset in the Amazon SageMaker catalog is approved by the data owner, Amazon SageMaker Unified Studio publishes an event in Amazon EventBridge in your account along with all the necessary information in the payload that enables you to create the access grants between the source and the target. When you receive this event, you can trigger a custom handler which can use the information in the event to create necessary grants or permissions. After you have granted the access, you can report back and update the status of the subscription in Amazon SageMaker Unified Studio so that it can notify the user(s) who subscribed to the asset that they can start consuming the asset.

Metadata enforcement rules for subscription requests

The metadata enforcement rules for subscription requests in Amazon SageMaker Unified Studio strengthen data governance by enabling domain unit owners to establish clear metadata requirements for data consumers, streamlining access requests and enhancing data governance.

The feature is supported in all the AWS commercial Regions where Amazon SageMaker Unified Studio is currently available.

Domain unit owners can complete the following procedure to configure metadata enforcement for subscription requests in Amazon SageMaker Unified Studio:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Govern -> Domain units** from the top navigation pane and then choose the domain or the domain unit that you want to work with.
3. Choose the **Rules** tab and then choose **Add**.
4. On the **Create required metadata form rule** page, do the following and then choose **Add rule**:
 - Specify a name for your rule.
 - Under **Action**, choose **Subscription request**.
 - Under **Required forms**, choose **Add metadata form**, choose a metadata form within the domain / domain unit that you want to add to this rule, and then choose **Add**. You can add up to 5 metadata forms per rule.
 - Under **Scope**, specify with which data entities you want to associate these forms. You can choose data products and/or data assets.
 - Under **Data asset types**, specify whether the rule applies across all asset types or limit it to selected asset types.
 - Under **Projects**, specify whether the required forms will be associated with data products and/or assets published by all projects or only selected projects in this domain unit. Also, check **Cascade rule to child domain units** if you want child domain units to inherit this requirement.

Fine-grained access control to data

In the current release of Amazon SageMaker Unified Studio, fine-grained access control of your data is supported so you can have granular access control over your sensitive data. You can control which project can access specific records of data within your data assets published to the Amazon SageMaker Unified Studio business data catalog. Amazon SageMaker Unified Studio supports row and column filters to implement fine-grained access control.

Use **row filters** to restrict access to specific rows based on the criteria you define. For example, if your table contains data for two regions (America and Europe) and you want to ensure that employees in Europe can only access data relevant to their region, you can create a row filter that

includes rows where the region is Europe (`region = 'Europe'`). This way, employees in Europe won't have access to America's data.

Use **column filters** to limit access to specific columns within your data assets. For example, if your table includes sensitive information such as Personally Identifiable Information (PII), you can create a column filter to exclude PII columns. This ensures that subscribers can only access non-sensitive data.

To utilize fine-grained access control, you can create row and column filters for your AWS Glue and Amazon Redshift assets in Amazon SageMaker Unified Studio. When you receive a subscription request to access your data assets, you can approve it by applying the appropriate row and column filters. Amazon SageMaker Unified Studio ensures that the subscriber can only access the rows and columns permitted by the filters you applied at the time of subscription approval.

Topics

- [Limitations](#)
- [Create row filters in Amazon SageMaker Unified Studio](#)
- [Create column filters in Amazon SageMaker Unified Studio](#)
- [Delete row or column filters in Amazon SageMaker Unified Studio](#)
- [Edit row or column filters in Amazon SageMaker Unified Studio](#)
- [Grant access with filters in Amazon SageMaker Unified Studio](#)

Limitations

When configuring row or column level filters for fine-grained access control, filtering on columns whose name contains special characters impacts which compute types can access the data.

- In cases where the column name contains special characters, adding an Asset Filter will automatically add double quotes “ ” around the column name to escape the special characters.

As a result, the asset is not accessible by data processing compute engines such as EMR-EC2, EMR-Serverless, or Glue-ETL. This asset is still accessible by other compute engines.

To remove this limitation, either remove the filters on the column names containing special characters or rename the column to remove the special characters and recreate the filter.

Create row filters in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio allows you to create row filters that you can use when approving subscriptions to make sure that the subscriber can only access rows of data as defined in the row filters. To create a row filter, follow the steps below:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. Make sure you are on the **Inventory** tab, then choose the name of the asset that you want to create a column filter for. You can add column filters if your data asset in Amazon SageMaker Unified Studio is of type AWS Glue table, Amazon Redshift table, or Amazon Redshift view. You are then brought to the asset details page.
5. On the asset detail page, go to the **Asset filters** tab and then choose **Add asset filter**.
6. Configure the following fields:
 - **Name** - the name of the filter
 - **Description** – the description of the filters
7. Under filter type, choose **Row filter**.
8. Under row filter expression, provide one or more expressions for row filter.
 - Choose a column from the **Column** dropdown.
 - Choose an operator from the **Operator** dropdown.
 - Enter a value in the **Value** field.
9. To add another condition to your filter expression, choose **Add condition**.
10. When using multiple conditions in the row filter expression, choose **And** or **Or** to link the conditions.
11. Select an option to indicate whether or not the filter contains sensitive values that you want to hide from approved subscribers.
12. Choose **Create asset filter**.

Create column filters in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables you to create column filters that you can use when approving subscriptions to make sure that the subscriber can only access columns of data as defined in the column filters. To create a column filter, follow the steps below:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. Make sure you are on the **Inventory** tab, then choose the name of the asset that you want to create a column filter for. You can add column filters if your data asset in Amazon SageMaker Unified Studio is of type AWS Glue table, Amazon Redshift table, or Amazon Redshift view. You are then brought to the asset details page.
5. On the asset detail page, go to the **Asset filters** tab and then choose **Add asset filter**.
6. Configure the following fields:
 - **Name** – the name of the filter
 - **Description** – the description of the filters
7. Under filter type, choose **Column**.
8. Select the columns you want to include in the filters using the check boxes for the columns in the data asset.
9. Choose **Create asset filter**.

Delete row or column filters in Amazon SageMaker Unified Studio

To delete a row or a column filter, follow the steps below:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.

4. Make sure you are on the **Inventory** tab, then choose the name of the asset where you want to delete a row or a column filter.
5. On the asset details page, go to the **Asset filters** tab and then choose the name of the filter that you want to delete.
6. Choose **Actions, Delete** and then confirm the deletion.

 **Note**

You can delete a filter only if it is not being used in active subscriptions.

Edit row or column filters in Amazon SageMaker Unified Studio

To edit a row or a column filter, follow the steps below:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Under **Project catalog** in the left side navigation, choose **Assets**.
4. Make sure you are on the **Inventory** tab, then choose the name of the asset that contains the filter that you want to edit.
5. On the asset detail page, go to **Asset filters** tab and then choose the name of the filter that you want to edit.
6. You can edit the following fields:
 - **Name** – the name of the filter
 - **Description** – the description of the filters
7. If you're editing a row filter, you can update the row filter expression.
8. If you're editing a column filter, you can add or remove the columns selected in the filter.
9. After you have made the changes, choose **Edit asset filter**.

Note

If you edit a filter that is being used in active subscriptions, Amazon SageMaker Unified Studio will automatically update the permissions granted to the subscriber projects. This means that the subscribers will only be able to access the rows or columns as defined in the updated filter, ensuring that your data access policies are consistently enforced.

Grant access with filters in Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio enables fine-grained access control by translating the defined row and column filters into appropriate grants for AWS Lake Formation and Amazon Redshift. Below is an explanation of how Amazon SageMaker Unified Studio materializes these filters for both AWS Glue tables and Amazon Redshift.

AWS Glue tables

When a subscription to an AWS Glue table with row and/or column filters is approved, Amazon SageMaker Unified Studio materializes the subscription by creating grants in AWS Lake Formation with Data Cell Filters, ensuring that the members of the subscriber project are only able to access the rows and columns they are allowed to access based on the filters applied to the subscription.

Amazon SageMaker Unified Studio first translates the row and columns filters applied in Amazon SageMaker Unified Studio to AWS Lake Formation Data Cell Filters. If multiple row and columns filters are used, Amazon SageMaker Unified Studio unions all the columns and all the row filter conditions to compute effective permissions at both row and column level. Amazon SageMaker Unified Studio then creates a single AWS Lake Formation data cell filter using effective row and column permissions.

After the data cell filter is created, Amazon SageMaker Unified Studio shares the subscribed table with the subscriber project by creating read-only (SELECT) permissions in AWS Lake Formation using this data cell filter.

Amazon Redshift

When a subscription to an Amazon Redshift table/view with row and/or column filters is approved, Amazon SageMaker Unified Studio materializes the subscription by creating scoped-down late binding views in Amazon Redshift, ensuring that the members of the subscriber project are only

able to access the rows and columns they are allowed to access based on the row and column filters applied to the subscription.

Amazon SageMaker Unified Studio first translates the row and columns filters applied to a subscription in Amazon SageMaker Unified Studio to an Amazon Redshift late binding view. If multiple row and columns filters are used, Amazon SageMaker Unified Studio unions all the columns and all the row filter conditions from to compute effective permissions at both row and column level. Amazon SageMaker Unified Studio then creates the late binding view using effective row and column permissions.

After the late binding view is created, Amazon SageMaker Unified Studio shares this view with the members of subscriber project by creating read-only (SELECT) permissions in Amazon Redshift.

Amazon SageMaker Lakehouse

Amazon SageMaker Lakehouse unifies your data across Amazon S3 data lakes and Amazon Redshift data warehouses, helping you build powerful analytics, machine learning (ML), and generative AI applications on a single copy of data. Amazon SageMaker Lakehouse provides integrated access controls and open-source [Apache Iceberg](#) for data interoperability and collaboration. With Amazon SageMaker Lakehouse, you can build an open lakehouse on your existing data investments, without changing your data architecture.

Amazon SageMaker Lakehouse provides the following key capabilities.

- **Unified data access** - With Amazon SageMaker Lakehouse, you can query and access data across Amazon S3 data lakes, Amazon Redshift data warehouses, and other sources using [Apache Iceberg](#) compatible tools and engines. This includes AWS services such as Amazon Athena, Amazon Redshift, Amazon EMR, Amazon SageMaker AI, as well as third-party engines, all of which you can use to query your data in-place.
- **Integrated access control** - Amazon SageMaker Lakehouse provides integrated fine-grained access control to your data. This means that you can define permissions and consistently apply them across all analytics and ML tools and engines, regardless of the underlying storage formats or query engines used.
- **Open source compatibility** - Amazon SageMaker Lakehouse leverages open-source [Apache Iceberg](#), enabling data interoperability across various Apache Iceberg compatible query engines and tools. This gives you the flexibility to choose your preferred tools and engines.

In this chapter:

- [How Amazon SageMaker Lakehouse works](#)
- [Amazon SageMaker Lakehouse key components](#)
- [Data connections in Amazon SageMaker Lakehouse](#)
- [Getting started with Amazon SageMaker Lakehouse](#)
- [Adding data sources in Amazon SageMaker Lakehouse](#)
- [Publishing data in Amazon SageMaker Lakehouse](#)

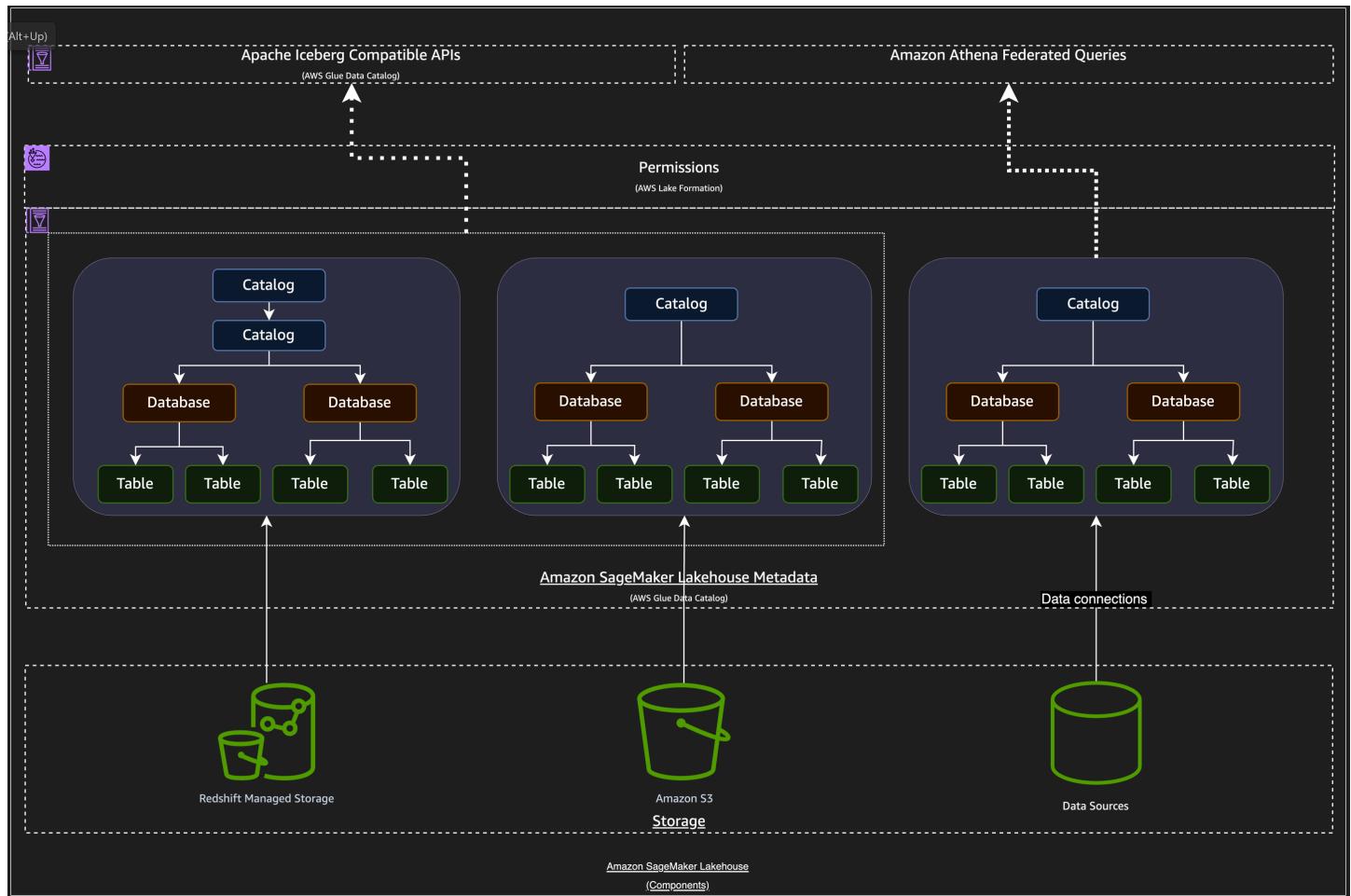
How Amazon SageMaker Lakehouse works

Amazon SageMaker Lakehouse is accessible from Amazon SageMaker Unified Studio. It organizes data from various sources into logical containers called catalogs. Each catalog represents data from existing sources like Amazon Redshift data warehouses, Amazon S3 data lakes, databases, or enterprise applications. You can also create new catalogs in the lakehouse to store data in S3 or Redshift Managed Storage (RMS).

You can access the data as Apache Iceberg tables and query it using any Iceberg-compatible engine, such as [Apache Spark](#), Amazon Athena, or Amazon EMR. Additionally, these catalogs are mounted as databases in Amazon Redshift, so you can connect and analyze your lakehouse data using SQL tools.

Amazon SageMaker Lakehouse is built on AWS Glue Data Catalog and AWS Lake Formation in your AWS account. With Amazon SageMaker Lakehouse, you can access and query your existing data in Amazon Redshift data warehouses and store new data in RMS from any Apache Iceberg compatible engine.

The following diagram shows how Amazon SageMaker Lakehouse works. Catalogs contain databases, which then contain tables. Types of storage sources for data that goes into catalogs include Redshift Managed Storage, Amazon S3, and data sources that you connect to with data connections.



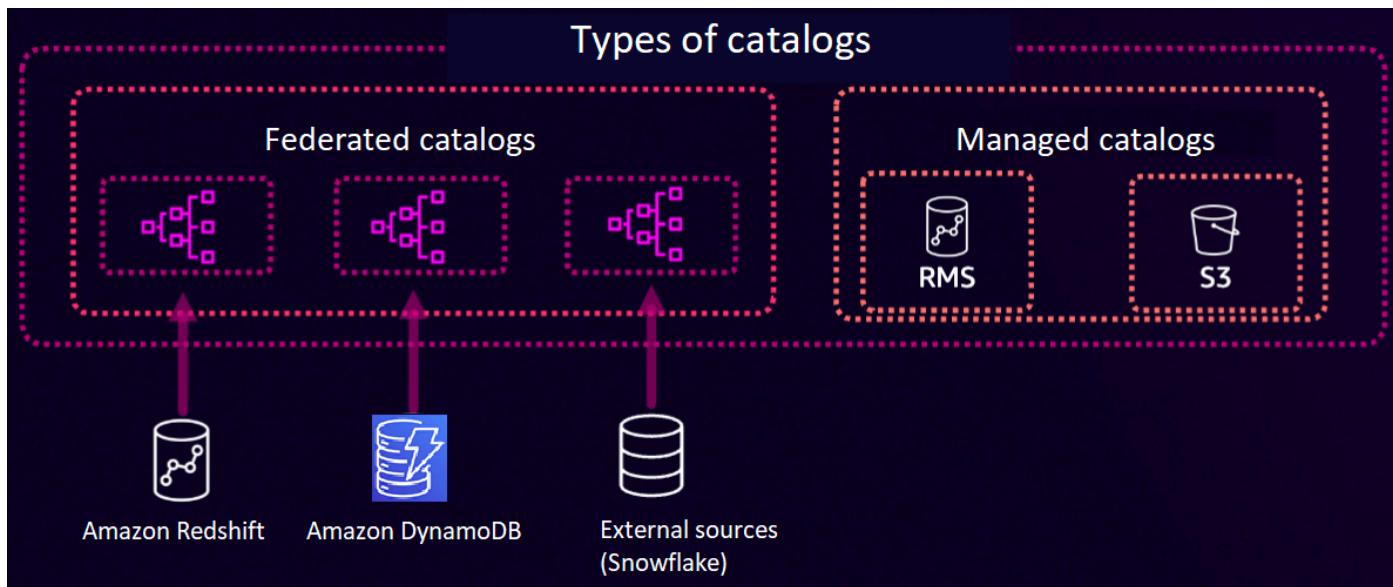
Amazon SageMaker Lakehouse key components

Amazon SageMaker Lakehouse has the following key components.

Catalog

A catalog is a logical container that organizes objects from a data store, such as schemas, tables, views, or materialized views such as from Amazon Redshift. You can create nested catalogs to mirror the hierarchical structure of your data sources within SageMaker AI Lakehouse.

There are two types of catalogs in Lakehouse: federated catalogs and managed catalogs. A federated catalog mounts existing data sources you add to Lakehouse. A federated catalog can bring existing data in data sources such as Amazon Redshift, Amazon DynamoDB, and Snowflake. A managed catalog refers to a new catalog you create using Lakehouse. A managed catalog manages data using RMS or S3, as shown in the following diagram.



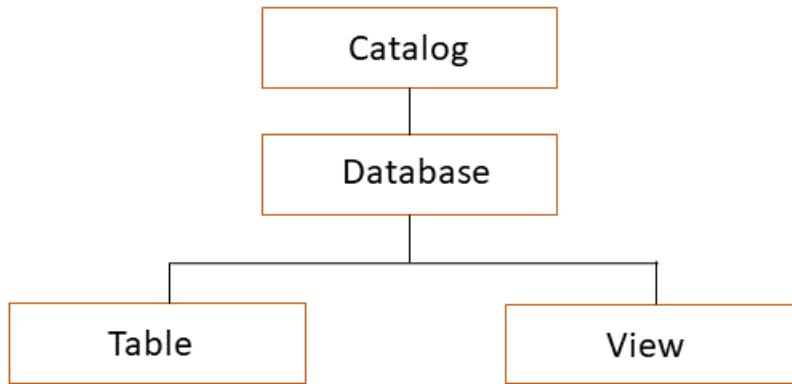
Database

Databases organize metadata tables in a catalog in Amazon SageMaker Lakehouse.

Table/View

Tables and views are database objects that define how to access and represent the underlying data. They specify details such as schema, partitions, storage location, storage format, and the SQL query required to access the data.

The following is a diagram of how catalogs, databases, tables/views work in Lakehouse.



Storage

You can read and write data into Amazon S3 or Redshift Managed Storage (RMS) based on the storage type you choose to store data in the lakehouse.

Data connections in Amazon SageMaker Lakehouse

Amazon SageMaker Lakehouse provides a unified approach to managing data connections across AWS services and enterprise applications. These connections provide a consistent experience for creating, testing, and exploring data sources, regardless of the underlying data platform.

Capabilities

With Amazon SageMaker Lakehouse connections, you can do the following:

- Create connections to a variety of data sources, including databases and data lakes
- Manage data connections in a single place
- Test the connectivity of your data sources to ensure they are working as expected
- Browse the metadata and preview the data from your connected sources
- Reuse the same connection across different AWS services like AWS Glue, Amazon Athena and Amazon SageMaker AI
- Manage credentials using AWS Secrets Manager
- Authenticate using basic authentication methods such as OAuth2 and IAM

Supported data sources

Amazon SageMaker Lakehouse connections support several popular data sources, including the following:

Supported Data Sources

Data Source	Type
Google BigQuery	Database
Amazon DocumentDB	Database
Amazon DynamoDB	Database
Amazon Redshift	Database
MySQL	Database

Data Source	Type
PostgreSQL	Database
SQL Server	Database
Snowflake	Database
Oracle	Database

 **Note**

Amazon SageMaker Lakehouse currently supports lowercase table, column, and database names. For optimal experience in Amazon SageMaker Unified Studio, ensure that all database identifiers are in lowercase.

Using Amazon SageMaker Lakehouse connections

After you've created an Amazon SageMaker Lakehouse connection, you can use it in various AWS services:

- Amazon SageMaker Unified Studio : Browse metadata, preview sample data, and run SQL queries against the connected data.
- AWS Glue: Use the connection for ETL jobs and crawlers.
- Amazon Athena: Query data directly using Athena's federated query capabilities. For more information, see [Register federated catalogs in Amazon Athena](#).
- Amazon SageMaker AI: Access data for building machine learning models.

Understanding created AWS resources

When you create a connection in Amazon SageMaker Unified Studio, several resources are created in your AWS account(s) behind the scenes. These resources can include:

- AWS Glue connection - A connection object is created in the AWS Glue crawler. This stores the core connection information and is used by various AWS services.

- Athena data catalog - For connections that will be used with Athena , an Athena data catalog is created. This allows Athena to query the external data source.
- AWS Glue data catalog entries - Databases, tables, and schemas from your external data source are registered in the Data Catalog. This enables AWS services to understand the structure of your external data.
- Lambda (for Athena Federated Query) - For some data sources, a Lambda function is created to facilitate federated queries. This function acts as a bridge between Athena and the external data source.

To view these resources, access the respective AWS service consoles (AWS Glue, Athena, IAM, etc.) in the AWS account associated with your Amazon SageMaker Unified Studio project.

In these consoles, look for resources with names that include your Amazon SageMaker Unified Studio project ID or connection name.

For more information about how to create a data connection and explore a connected data source, see [???](#).

Getting started with Amazon SageMaker Lakehouse

This tutorial covers information to help you get started using Amazon SageMaker Lakehouse as a user. If you are new to Amazon SageMaker Lakehouse, start by reading [???](#). If you are new to Amazon SageMaker Unified Studio, start by reading the concepts and terminology in [Terminology and concepts](#).

Topics

- [Prerequisites](#)
- [Create a project](#)
- [Browse data](#)
- [Upload data](#)
- [Query data](#)

Prerequisites

- Your administrator must grant you access to [Amazon SageMaker Unified Studio](#).

- If you don't have access to it, contact your administrator. For more information, see [???](#).
- You must have an Amazon SageMaker Unified Studio project and with the proper project membership role.

If you don't have proper access to a project, contact your administrator. To view your project membership role, choose **Actions** on the top right corner of the project overview page, then choose **Manage members**. You will see your membership role in the **Role** column.

Create a project

You can create a project from a project profile, which defines a template for projects in your domain. To use Amazon SageMaker Lakehouse, your project must be created using either [Data analytics and AI-ML model development](#) or [SQL analytics](#) project profile. For more information about creating a project, see [???](#) from Amazon SageMaker Unified Studio User Guide.

When using Amazon SageMaker Unified Studio, you can create the following resources in the lakehouse:

1. Databases in AWS Glue Data Catalog

Amazon SageMaker Lakehouse is implemented on AWS Glue and AWS Lake Formation in your AWS account.

2. A catalog to store data in Redshift Managed Storage (RMS) format

You will create a catalog in RMS format. To view the catalog, navigate to the AWS Lake Formation console at <https://console.aws.amazon.com/lakeformation/>, you should be able to see the catalog from the **Catalogs** list.

3. Provisioning permissions

You will create an IAM role when you create a project. Each project has a dedicated IAM role. This IAM role has permission to the resources that are created from this project. The Amazon Resource Name (ARN) of this IAM role is visible from **Project details** section of the **Project overview** page.

Browse data

You can browse data in Amazon SageMaker Lakehouse by completing the following steps.

To browse data

1. Choose a project to view the data.
2. On project page, from the left navigation, choose **Data**. This opens the **Data** explorer in the middle of the page.

The **Data** explorer includes: **Lakehouse**, **Redshift**, and **S3**.

3. Expand **Lakehouse** to view catalogs, databases, tables.

Upload data

You can upload data in CSV or JSON format to a catalog. To upload data, follow the instructions in [???](#).

After uploading data is complete, you will see the table listed within the database under **AwsDataCatalog**.

Query data

You can query data using supported query editor.

To query data

1. On **Lakehouse**, choose **AwsDataCatalog** on top. Expand the catalog to view the list of databases. Choose a database.
2. From a selected database, choose a table. Then choose the three dot menu to the right of the table to view supported tools for data query.
3. Choose **Query with Athena**. This opens the **Data explorer** page where you can run SQL queries. You might find information in [SQL reference for Athena](#) helpful.
4. Choose **Query with Amazon Redshift**. This opens the **Data explorer** page where you can run SQL queries. You might find information in [Querying a database using the query editor v2](#) helpful.

To subscribe an asset, see [???](#).

To publish data to the catalog from the lakehouse inventory, see [???](#).

Adding data sources in Amazon SageMaker Lakehouse

Amazon SageMaker Lakehouse supports several data sources. If you are new to data connections in Amazon SageMaker Lakehouse, see [???](#).

In this topic:

- [Creating connections in Amazon SageMaker Lakehouse](#)
- [Uploading data](#)
- [Creating a catalog](#)
- [Adding existing databases and catalogs using AWS Lake Formation permissions](#)
- [Amazon S3 tables integration](#)

Creating connections in Amazon SageMaker Lakehouse

Amazon SageMaker Unified Studio provides an interface for managing and utilizing data connections across various AWS services and external data sources. With Amazon SageMaker Unified Studio, you create, configure, and manage connections to databases, data warehouses, and applications all from a single platform. Amazon SageMaker Unified Studio allows you to explore your connected data sources, preview sample data, and seamlessly use these connections in SQL queries and Spark notebooks without having to switch between different interfaces or manage complex connection details manually.

Access the data explorer in a project

1. Open your web browser and navigate to Amazon SageMaker Unified Studio.
2. Enter your corporate credentials (usually integrated with Amazon IAM Identity Center).
3. After successful authentication, you'll be directed to the Amazon SageMaker Unified Studio home page. On the home page, you'll see a list of projects you have access to. Select the project you want to work with by clicking on its name.
4. From the dropdown menu, select the **Data** or **Data Management** option. This will open the Data section of the project overview page. In this data explorer, you can see a tree-like structure representing your data sources.

Create a new connection to add data sources

To add a new data source

1. In the data explorer, select the + button. Click this button to start adding a new data source.
2. In the modal, select **Add connection**. You'll be presented with a gallery of connector options. Select the connector you need. For supported data sources, see [???](#).

 **Note**

Amazon SageMaker Lakehouse currently supports lowercase table, column, and database names. For optimal experience in Amazon SageMaker Unified Studio, ensure that all database identifiers are in lowercase.

3. You must configure your connector details. For example, if you choose to use a DynamoDB connection (preview), fill in the required fields, which can include:
 - Name: A unique identifier for this connection in Amazon SageMaker Unified Studio.
 - Description (optional): A description of the connection.

 **Note**

Each supported data source can have different parameters for the connection. Contact your administrator if you need them.

To see your DynamoDB tables displayed in Amazon SageMaker Lakehouse after you add the connection, your administrator must grant you access through resource policies in the Amazon DynamoDB console.

To grant access to a DynamoDB table, your administrator can complete the following steps.

1. Sign in to the AWS Management Console and open the Amazon DynamoDB console at <https://console.aws.amazon.com/dynamodb/>.
2. On the left navigation of the DynamoDB console, choose **Tables**.
3. From the **Tables** page, choose the table to add access to.

4. On the details page of the selected table, choose **Permission**.
5. On the **Resource-based policy for table** section, update the policy with the project role ARN in Condition.

 **Note**

You can find the project ARN on the Page details page in Amazon SageMaker Unified Studio.

The following is an example policy. It allows access of the IAM role named `datazone_user_role_projectid` to perform the allowed actions (Query, Scan, DescribeTable, PartiQLSelect) on the specified DynamoDB table. Administrators should choose to allow or deny the set of actions.

```
{  
    "Sid": "Statement1",  
    "Effect": "Allow",  
    "Principal": "*",  
    "Action": [  
        "dynamodb:Query",  
        "dynamodb:Scan",  
        "dynamodb:DescribeTable",  
        "dynamodb:PartiQLSelect"  
    ],  
    "Resource": "arn:aws:dynamodb:region:account:table/table_name",  
    "Condition": {  
        "ArnEquals": {  
            "aws:PrincipalArn": "arn:aws:iam::region:role/datazone_user_role_projectid"  
        }  
    }  
}
```

Explore a connected data source

After you have connected your data source, you can explore the data source in the data explorer.

1. After your connection is created, return to the data explorer.
2. You should now see your new connection listed in **Lakehouse**.

3. Expand the new connection to view available databases.
4. Expand a database to explore its schema.
5. You can select a table name to view more details about that table, such as Schema details and a list of tables. You can then examine the tables themselves by selecting a table.
6. You will be able to see tabs for **Columns** and **Sample data**. In the **Columns** view, you can view a list of columns in the table, as well as the data types for each column. In the **Sample data** view, you can see the rows of data from the table and use built-in sorting and filtering options to explore the data.

Authentication and tagging for creating connections

You administrator must create credentials and configure the secret tags for you before you create a connection.

Credentials

When creating a connection, if you choose a data source that requires the credentials for **Authentication**, contact your administrator because they must create and provide these credentials. There are two types of the credentials:

- User name and password
- AWS Secrets Manager

Secret tags

- To ensure the secret can only be used for a particular project, your administrator must tag with the AmazonDataZoneProject tag key and the value will be projectId.
- To use the secret across multiple projects, your administrator must tag the secret with for-use-with-all-datazone-projects = true.

Uploading data

You can upload data to Amazon SageMaker Lakehouse.

To upload data

1. On the **Data** section in the middle of the project page, choose **+** on the top. This opens **Add data source** on the right.
2. On **Add data source**, choose **Upload data**.
3. Choose **Click to upload** or drag and drop a CSV or JSON file. Complete the information in the form.
4. Choose **Upload data**.

Creating a catalog

You can create a catalog for your Redshift Managed Storage (RMS) objects.

To create a catalog

1. On the **Data** explorer in the middle of the project page, choose **+** on the top. This opens **Add data source** on the right.
2. On **Add data source**, choose **Create catalog**. Enter a name for your catalog.
3. Choose **Create**.

Adding existing databases and catalogs using AWS Lake Formation permissions

You can add existing databases and catalogs to Amazon SageMaker Lakehouse.

To add existing databases and catalogs using AWS Lake Formation permissions

1. Sign in to Amazon SageMaker Unified Studio by using the link your administrator gave you. If you don't have access to it, contact your administrator.
2. Choose a project to open the project page.
3. On the left navigation, choose **Project overview**. On **Project details**, copy the project role ARN.
4. Open the AWS Lake Formation console at <https://console.aws.amazon.com/lakeformation/>.
5. On the left navigation, from **Data catalog**, choose **Catalogs**.

6. On the **Catalogs** list view, choose a catalog you want to add to Amazon SageMaker Lakehouse. From **Actions** on the right, choose **Grant**.
7. On the **Grant data lake permissions** page, choose **IAM users and roles** from **Principals**. Paste the IAM role you copied in the step 3.
8. On **Catalog permissions**, choose **Super user**. Choose **Grant**.

After you complete all the steps successfully, go back to the project page in Amazon SageMaker Unified Studio. You should see the Lake Formation catalog added to your lakehouse.

Amazon S3 tables integration

Amazon SageMaker Lakehouse unifies all your data across Amazon S3 data lakes, Amazon Redshift data warehouses, and third-party data sources without having to copy data. Amazon S3 Tables delivers the first cloud object store with built-in Apache Iceberg support. Amazon SageMaker Lakehouse integrates with Amazon S3 Tables so you can access S3 Tables from AWS analytics services, such as Amazon Redshift, Amazon Athena, Amazon EMR, AWS Glue, or Apache Iceberg-compatible engines (Apache Spark or PyIceberg).

Amazon SageMaker Lakehouse integration with Amazon S3 Tables helps you secure analytic workflows by joining data from Amazon S3 Tables with sources, such as Amazon Redshift data warehouses, third-party, and federated data sources (Amazon DynamoDB or PostgreSQL). SageMaker Lakehouse also enables centralized management of fine-grained data access permissions for S3 Tables and other data, and consistently applies them across all engines. To get started, complete the steps in the following sections.

Prerequisites - complete all the steps in the [Getting started with Amazon SageMaker Lakehouse](#).

Enable Amazon S3 Integration

1. Navigate to the [Amazon S3 console](#). In the left navigation pane, choose **Table buckets**.
2. Choose **Create table bucket**.
3. On the **Create table bucket** page, enter a **Table bucket name** and select **Enable integration**.
4. Choose **Create table bucket**.
5. You will see confirmation when Amazon S3 completes integration of your table buckets with SageMaker Lakehouse.

Onboard S3 Tables in SageMaker Lakehouse

To provide access to S3 tables, complete the following steps:

1. Navigate to the [AWS Lake Formation](#) console.
2. In the left navigation pane, choose **Catalogs** and choose **S3tablescatalog**.
3. From **S3tablescatalog**, under **Objects**, choose the name of your newly created **table bucket**.
4. From the **Actions** menu, select **Grant**.
5. In the **Grant permissions**, under IAM users and roles, select your Amazon SageMaker Unified Studio Project role. To grant full access, under **Catalog Permissions > Grant**, select **Super user**.

Create S3 Table and add data in SageMaker Lakehouse

1. Navigate to Amazon SageMaker Unified Studio, and select the project.
2. From the **Build** menu, select **Query Editor**, and ensure you have **Athena** selected in **Connections**.
3. Create a database using SQL.

```
CREATE DATABASE "s3tablescatalog/<Your Bucket Name>".<YourDBName>;
```

4. Create an S3 table using SQL.

```
CREATE TABLE "s3tablescatalog/<Your Bucket Name>".<YourDBName>.<YourTableName>
( c_salutation string,
  c_login string,
  c_first_name string,
  c_last_name string,
  c_email_address string)
TBLPROPERTIES (
  'table_type'='ICEBERG' );
```

5. Add data using SQL.

```
INSERT INTO "s3tablescatalog/<Your Bucket Name>".<YourDBName>.<YourTableName>
VALUES('Dr.', '1381546', 'Joyce', 'Deaton', 'Joyce.Deaton@qhtrwert.edu');
```

You can now use the following integrated analytics services:

- [Amazon Athena](#) - create databases, tables, query and add data in S3 Tables.
- [Amazon Redshift](#) - query data from S3 Tables.
- [Amazon EMR](#) - create table, namespace, query and add data in S3 Tables.
- [AWS Glue](#) - create table, namespace, query and add data in S3 Tables.
- [AWS Lake Formation](#) - grant fine-grained permissions for S3 table catalogs, databases, tables, columns, and cells.

 **Note**

Access to S3 Tables with SageMaker Lakehouse is available in the [AWS Regions](#) where S3 Tables are available.

Publishing data in Amazon SageMaker Lakehouse

After you have added data in Amazon SageMaker Lakehouse, you can publish the data to share it with other users in Amazon SageMaker Unified Studio. Data that is published is viewable as an asset in the project catalog and the Amazon SageMaker catalog, and other users can create subscription requests in the Amazon SageMaker catalog to include that data in their projects.

To publish data in Amazon SageMaker Lakehouse, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the data that you want to publish in Amazon SageMaker Lakehouse. To do this, use the center menu at the top of the landing page and choose **Browse all projects**, then choose the name of the project that you want to navigate to.
3. In the center menu, choose **Data**. This takes you to the Data page.
4. Do either of the following:
 - If you want to publish a regular AWS Glue table, expand the catalog in the data navigation to view the list of databases in Amazon SageMaker Lakehouse, then choose a database that

contains the asset that you want to publish. Choose this table from the selected database and then proceed to the rest of the steps in this procedure to publish this table to the catalog.

- If you want to publish an Amazon S3 table to the catalog, you must first complete the following steps to create a data source for the S3 Tables catalog and schedule its run job. Then you can proceed to the rest of the steps in this procedure to publish the S3 table to the catalog.
 - Navigate to **Data sources** and then choose **Create data source**.
 - On the **Step 1: Define source** page, specify the name for this data source, then under **Data source type** - choose **AWS Glue (Lakehouse)**, under **Data Selection** - choose **Enter the catalog name** and then specify the name of your S3 tables catalog (`s3tablescatalog/<catalog name>`), then choose your database from that catalog (use the drop down menu), and then choose **Next**.
 - On the **Step 2: Add details** page, leave all the default settings and choose **Next**.
 - On the **Step 3: Set up schedule** page, choose a run preference and then choose **Next**.
 - On the **Step 4: review** page, review your selections and then choose **Create**.

Once the data source for the S3 tables catalog is created and run, you can proceed with the rest of the steps below to locate your S3 table and publish it to the catalog.

5. Expand the **Actions** menu, then choose **Publish to catalog**.
6. Confirm the action in the pop-up window by choosing **Publish to catalog**.

Amazon SageMaker Unified Studio then fetches metadata for the asset. After a few minutes, the metadata is fetched and a success message appears.

7. (Optional) Choose **View details** to view the asset in the project catalog.

When it is successfully published you can view it in the **Assets** section of the project catalog and users in other projects can subscribe to it from the Amazon SageMaker catalog.

You can use the project catalog to re-publish the data if you make changes, or to unpublish the data from Amazon SageMaker catalog. For more information, see [the section called “Data inventory and publishing”](#).

Compute

On a project's **Compute** page in Amazon SageMaker Unified Studio, you can view compute information and add compute resources such as Amazon Redshift and Amazon EMR Serverless clusters to your project. Amazon SageMaker Unified Studio supports different kinds of compute resources:

- **Data warehouse:** This includes Amazon Redshift Serverless workgroups and Amazon Redshift provisioned clusters. Workgroups are a collection of compute resources that you can use to run data warehousing queries and engineering notebooks without managing underlying infrastructure. Clusters are scalable compute environments that enable the processing and analysis of large datasets. For more information, see [the section called "Amazon Redshift"](#).
- **Data processing:** This includes connections to Amazon EMR on EC2 clusters and EMR Serverless applications, and Glue ETL computes. For more information, see the following links:
 - [the section called "Amazon EMR on EC2"](#)
 - [the section called "EMR Serverless"](#)
 - [the section called "Glue ETL"](#)
- **HyperPod clusters:** In Amazon SageMaker Unified Studio, you can launch machine learning workloads on Amazon SageMaker AI HyperPod clusters. For more information, see [the section called "HyperPod clusters"](#).
- **Spaces:** Spaces are used to manage the storage and resource needs of applications running on JupyterLab. On the **Spaces** tab of the **Compute** page, you can view information about your JupyterLab environment in Amazon SageMaker Unified Studio, such as the EBS volume and the status of the IDE. For more information about spaces, see [IDE spaces](#).
- **MLflow tracking servers:** MLflow tracking servers make it possible to use MLflow in Amazon SageMaker Unified Studio to create, manage, analyze, and compare machine learning experiments. For more information, see [the section called "Track experiments using MLflow"](#).
- **Workflow environments:** Use a workflow environment to share scheduled workflows with other project members. For more information, see [the section called "Create a workflow environment"](#).

 **Note**

Adding a serverless or cluster compute connection adds the compute resource to the project space, so all project members can access it.

Amazon Redshift compute connections in Amazon SageMaker Unified Studio

You can connect to Amazon Redshift Serverless workgroups and Amazon Redshift clusters in Amazon SageMaker Unified Studio.

Amazon Redshift Serverless workgroups are a collection of compute resources that you can use to run data warehousing queries and engineering notebooks without managing underlying infrastructure. These are especially useful in environments where query patterns are unpredictable or workloads fluctuate.

Amazon Redshift clusters are scalable compute environments that enable the processing and analysis of large datasets. They are optimized for running SQL-based queries on data warehouses, making them ideal for structured data analytics and reporting.

Gaining access to Amazon Redshift resources

To add Amazon SageMaker Unified Studio connections to existing compute resources, you must get access information from the admin that owns the resources. To do this, first get your project ID from the **Project overview** page of the project you want to add resources to. Then, send the project ID to the owner of the Amazon Redshift resources. The Amazon Redshift admin uses the project ID to complete some steps so that you receive access details from them, and then you can input the access information in Amazon SageMaker Unified Studio.

You and the admin must complete different steps depending on whether the resources are in the same account as the account you are accessing Amazon SageMaker Unified Studio in.

Note

If you want to query the Amazon Redshift resources using JupyterLab within Amazon SageMaker Unified Studio, the Amazon Redshift resource must use the same VPC as the Amazon SageMaker Unified Studio project. If the Amazon SageMaker Unified Studio project uses a different VPC than the Amazon Redshift resource you want to gain access to, you and your admin must complete additional steps to connect the VPCs before you can use JupyterLab to query. You can still query using the Data page of your project if you are using different VPCs. For more information, see [VPC to VPC connectivity](#) and [Connect VPCs using VPC peering](#).

Gaining access to resources in the same account

In some cases, the Amazon Redshift resource you want to add to your Amazon SageMaker Unified Studio project might be in the same account as your project.

For compute resources in the same account as your Amazon SageMaker Unified Studio project, complete the following steps:

1. Send the Amazon Redshift admin the project ID. This can be found on the [Project overview](#) page of your Amazon SageMaker Unified Studio project.
2. The admin then adds 1 of the following tags to the Amazon Redshift cluster or workgroup that you want to add to Amazon SageMaker Unified Studio.
 - Option 1: Add a tag to allow only a specific Amazon SageMaker Unified Studio project to access it: `AmazonDataZoneProject=projectID`.
 - Option 2: Add a tag to allow all Amazon SageMaker Unified Studio projects in this account to access it: `for-use-with-all-datazone-projects=true`.
3. The admin then must send you a username and password for a database user that has access to the compute resources.

You can then use the username and password to add the compute connection in Amazon SageMaker Unified Studio. For more information, see [the section called “Connecting to an existing Amazon Redshift resource”](#).

Gaining access to resources in a different account

In some cases, the Amazon Redshift resource you want to add to your Amazon SageMaker Unified Studio project might be in a different AWS account than your project.

For compute resources in a different account, complete the following steps:

1. Send the Amazon Redshift admin the following information from the [Project overview](#) page of your Amazon SageMaker Unified Studio project:
 - The Amazon SageMaker Unified Studio project role ARN.
 - The Amazon SageMaker Unified Studio project ID.
 - The Amazon SageMaker Unified Studio project domain ID.

2. The admin must create an access role for Amazon SageMaker Unified Studio that can be used to query Amazon Redshift.

An example Amazon Redshift access role for Amazon SageMaker Unified Studio is provided below:

```
# Sample permission policy of access role to query Redshift
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RedshiftQueryEditorConnectPermissions",
            "Effect": "Allow",
            "Action": [
                "redshift:GetClusterCredentialsWithIAM",
                "redshift:GetClusterCredentials",
                "redshift:DescribeClusters",
                "redshift>CreateClusterUser"
            ],
            "Resource": [
                "arn:aws:redshift:*:012345678912:cluster:*",
                "arn:aws:redshift:*:012345678912:dbname:*/**",
                "arn:aws:redshift:*:012345678912:dbuser:*/**"
            ]
        },
        {
            "Sid": "RedshiftServerlessQueryEditorConnectPermissions",
            "Effect": "Allow",
            "Action": [
                "redshift-serverless:GetCredentials",
                "redshift-serverless:GetWorkgroup",
                "redshift-serverless>ListTagsForResource"
            ],
            "Resource": [
                "arn:aws:redshift-serverless:*:012345678912:workgroup/*"
            ]
        },
        {
            "Sid": "SecretsManagerAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue",
```

```
        "secretsmanager:DescribeSecret"
    ],
    "Resource": [
        "secret_arn"
    ]
},
{
    "Sid": "sqlworkbench",
    "Effect": "Allow",
    "Action": [
        "sqlworkbench:*"
    ],
    "Resource": [
        "*"
    ]
}
]
```

The trust policy is as follows:

```
# trust policy of access role
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "project-role-arn"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "sts:ExternalId": "project-id"
                }
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "project-role-arn"
            }
        }
    ]
}
```

```
        },
        "Action": [
            "sts:SetSourceIdentity"
        ],
        "Condition": {
            "StringLike": {
                "sts:SourceIdentity": "${aws:PrincipalTag/datazone:userId}"
            }
        }
    },
    {
        "Effect": "Allow",
        "Principal": {
            "AWS": "project-role-arn"
        },
        "Action": "sts:TagSession",
        "Condition": {
            "StringEquals": {
                "aws:RequestTag/AmazonDataZoneProject": "project-id",
                "aws:RequestTag/AmazonDataZoneDomain": "domain-id"
            }
        }
    }
]
```

3. (Optional) If you want to use IAM credentials to access the Amazon Redshift resource, rather than an AWS Secrets Manager secret, the admin must add the following tag to the access role:

`RedshiftDbUser=Username`

4. The admin then needs to provide JDBC connection info in one of two ways:

- Use a Secrets Manager secret in the same account as the Redshift resource. The access role should have permission to read the secret value. For more information about the JSON format that should be used in the secret, see [JSON structure of a secret](#) in the AWS Secrets Manager User Guide.
- Use a temporary username and password. This is generated from the IAM access role credentials.
 - The `RedshiftDbUser` tag on the access role is required. This determines the federated database user within the databases for the Amazon SageMaker Unified Studio users. For

more information, see [Setting up principal tags to connect a cluster or workgroup from query editor v2](#) in the Amazon Redshift Management Guide.

5. The admin then sends you the following information:

- Access role ARN.
- JDBC URL. For example: `jdbc:redshift://default-workgroup.012345678912.us-west-2.redshift-serverless.amazonaws.com`. For more information about JDBC connections, see [Connecting to Amazon Redshift Serverless through JDBC drivers](#) and [Getting the JDBC URL](#) in the in the Amazon Redshift Management Guide..
- (Optional) AWS Secrets Manager secret ARN. For example: `arn:aws:secretsmanager:us-west-2:012345678912:secret:shared-rs-cluster-password-Ab1CDe`.

You can then use the access credentials and JDBC URL to add the compute connection in Amazon SageMaker Unified Studio. For more information, see [the section called “Connecting to an existing Amazon Redshift resource”](#).

Connecting to an existing Amazon Redshift resource

After you have gained access to an Amazon Redshift resource, you can add a connection to the compute resource in the Amazon SageMaker Unified Studio console. Complete the following steps to add a serverless or cluster compute to the project space:

1. Go to the **Compute** section of your project in Amazon SageMaker Unified Studio.
2. Select the **Data warehouse** tab.
3. Choose **Add compute**.
4. Choose **Connect to existing compute resources**, then choose **Next**.
5. Select the type of compute resource you want to add, then choose **Next**.
6. Under **Connection properties**, provide the JDBC URL or the compute you want to add. If the compute resource is in the same account as your Amazon SageMaker Unified Studio project, you can select the compute resource from a dropdown menu. For more information, see [the section called “Gaining access to Amazon Redshift resources”](#).
7. Under **Authentication**, provide the credential type you want to use to access the resource. The credential type must be one of the following options: Username and password, IAM credentials, AWS Secrets Manager.
8. Provide the credentials according to the authentication method you selected.

9. Under **Name**, input the name of the Amazon Redshift Serverless or Amazon Redshift Cluster you want to add.

10. Under **Description**, provide a description of the compute resource.

11. Choose **Add compute**. The Amazon SageMaker Unified Studio project Compute and Data pages then display information for that resource.

 **Note**

Some credentials provide more information than others on the Compute page. Using a username and password enables Amazon SageMaker Unified Studio to display more information for a resource.

Creating a new Amazon Redshift Serverless compute resource

You can create a new compute resource and add a connection to it in Amazon SageMaker Unified Studio. Complete the following steps to add a new Amazon Redshift Serverless compute connection to the project space:

1. Go to the **Compute** section of your project in the Amazon SageMaker Unified Studio .
2. On the **Data warehouse** tab, choose **Add compute**.
3. Choose **Create new compute resources**.
4. Select the type of compute resource you want to add.
5. Under **Compute name**, input a name for the Amazon Redshift Serverless resource you want to add.
6. Under **Description**, provide a description of the compute resource.
7. Set the base capacity, maximum capacity and database name.
8. Choose **Add compute**. The Amazon SageMaker Unified Studio project Compute and Data pages then display information for that resource.

Note

Some credentials provide more information than others on the Compute page. Using a username and password enables Amazon SageMaker Unified Studio to display more information for a resource.

Removing an Amazon Redshift compute connection

When you remove a compute connection in Amazon SageMaker Unified Studio, you delete the connection to the compute resource that your Amazon SageMaker Unified Studio project has without deleting the compute resource.

To remove a compute connection in Amazon SageMaker Unified Studio, complete the following steps:

1. Go to the **Compute** page of your project in the Amazon SageMaker Unified Studio.
2. Select the name of the compute connection you want to remove. You are then taken to the compute details page.
3. Choose **Actions > Remove compute**. A popup window appears asking you to confirm the removal.
4. To confirm the removal, input confirm in the text box provided.
5. Choose **Remove compute**.

This removes the Amazon SageMaker Unified Studio connection to the compute resource. You are then no longer able to access the compute resource in the Amazon SageMaker Unified Studio project, but the compute resource is not deleted.

Amazon EMR on EC2 connections in Amazon SageMaker Unified Studio

Whenever you are working with a project, you can manage that project's Amazon EC2 resources and view both monitoring and logging data for those resources. You can create and configure Amazon EMR on EC2 clusters, as well as terminate and remove those clusters. When clusters are running, data regarding their metrics is automatically sent to CloudWatch, while logging data is preserved in the Spark UI.

Adding a new Amazon EMR on EC2 cluster in Amazon SageMaker Unified Studio

As a data worker, you can make use of Amazon EMR on EC2 by adding existing or new Amazon EMR on EC2 clusters as compute instances to a project in the Amazon SageMaker Unified Studio Studio. Within a project, you can use both existing and new Amazon EMR on EC2 clusters.

Before you can create a new Amazon EMR on EC2 cluster, your admin must enable blueprints. On-demand creation isn't supported for Amazon EMR on EC2 in quick setup.

After your Admin has enabled blueprints:

1. From inside the project management view, select **Compute** from the navigation bar.
2. In the Compute panel, select the **Data processing** tab.
3. To create a new Amazon EMR on EC2 cluster, select the **Add compute** dropdown menu and then choose **New compute**.
4. In the **Add compute** modal, you can select the type of compute you would like to add to your project. Select **Create new compute resources**.
5. Select **Amazon EMR on EC2 cluster**.
6. The **Add compute** dialog box allows you to specify the name of the Amazon EMR on EC2 cluster, provide a description, and choose a release of EMR (such as EMR 7.5) that you want to install on your cluster.
7. After configuring these settings, select **Add compute**. After some time, your Amazon EMR on EC2 cluster will be added to your project.

Adding an existing Amazon EMR on EC2 cluster in Amazon SageMaker Unified Studio

As a data worker, you can make use of Amazon EMR on EC2 by adding existing or new Amazon EMR on EC2 clusters as compute instances to a project in the Amazon SageMaker Unified Studio Studio. Within a project, you can use both existing and new Amazon EMR on EC2 clusters.

Before you can connect to an Amazon EMR on EC2 cluster, you must complete the following prerequisites:

- Your Amazon SageMaker Unified Studio admin must enable blueprints. On-demand creation isn't supported for Amazon EMR on EC2 in quick setup. In addition, if you are connecting to an

Amazon EMR on EC2 cluster that is not runtime-role enabled, the admin must configure specific blueprints as described in the section below.

- You must have a project created in Amazon SageMaker Unified Studio. If you are connecting to an Amazon EMR on EC2 cluster that is not runtime-role enabled, you must create a project that includes specific blueprint configurations in the project profile.
- The admin that owns the Amazon EMR resource you want to connect to must complete a set of prerequisite steps to grant you access to the resource.

More details on each of these steps is found in the sections below.

Prerequisite steps for you and your Amazon SageMaker Unified Studio admin

Amazon EMR on EC2 clusters can be runtime-role enabled or not runtime-role enabled. You can connect to both kinds of Amazon EMR on EC2 clusters in Amazon SageMaker Unified Studio. However, to use clusters that are not runtime-role enabled, you and your Amazon SageMaker Unified Studio admin must prepare to use a project with specific configurations.

Note

If you are connecting to clusters that are runtime-role enabled, you can proceed to the section for prerequisite steps for Amazon EMR admins without completing the steps in this section.

- You can use runtime-role enabled clusters to specify different IAM roles for individual jobs or steps within a cluster, with fine-grained access control tailored to specific job needs.
- Clusters that are not runtime-role enabled have limited granular access control for jobs. Instead, all jobs on the cluster use the same set of permissions.

Amazon EMR clusters with runtime roles enabled are considered more secure because they allow for fine-grained access control at the job level, meaning each individual job running on the cluster can be assigned a specific IAM role with only the necessary permissions to access the data and resources it needs.

To prepare to use clusters that are not runtime-role enabled, complete the following additional steps:

Note

Amazon EMR clusters that are not runtime-role enabled must have in-transit encryption enabled in order to be connected to Amazon SageMaker Unified Studio. To ensure that the Amazon EMR cluster meets this requirement, verify with your Amazon EMR admin that the cluster has a security configuration with in-transit encryption enabled. For more information, see [Create a security configuration with the Amazon EMR console or with the AWS CLI](#) in the Amazon EMR Management Guide.

1. The Amazon SageMaker Unified Studio admin must configure the tooling configurations in the blueprints for a project profile so that **allowConnectionToUserGovernedEmrClusters** is set to **True** in the Amazon SageMaker Unified Studio management console. For more information, see the Amazon SageMaker Unified Studio Administrator Guide.
2. You create a project using the project profile that your admin modified in step 1.

For more information about runtime roles, see [Runtime roles for Amazon EMR steps](#) in the Amazon EMR Management Guide.

Note

For clusters without runtime roles, Amazon SageMaker Unified Studio cannot provide governance on the clusters, and applications running on these clusters will not be isolated between projects or honor fine-grained access control based on project data permissions. Additionally, all project resources are inaccessible to the cluster unless additional permissions are granted to the IAM instance profile role attached to the Amazon EC2 instance.

Prerequisite steps for Amazon EMR admins

Before you can add an existing Amazon EMR on EC2 resource to your project in Amazon SageMaker Unified Studio, the admin that owns that resource must grant access to you by completing the following steps:

Create an Amazon EMR access role with a trust policy

1. Get the project role ARN and project ID for the Amazon SageMaker Unified Studio project that you want to grant access to. Project members can get the project role ARN and project ID from the **Project overview** page in their project.

 **Note**

If the Amazon SageMaker Unified Studio project uses a different VPC than the Amazon EMR on EC2 cluster you want to grant access to, you must also get the project VPC information from the project member and complete additional steps to connect the VPCs. For more information, see [VPC to VPC connectivity](#) and [Connect VPCs using VPC peering](#).

2. Make sure that the EMR cluster you want to grant access to has an instance profile role with the `sts:AssumeRole` permission on the runtime role. For more information, see [Runtime roles for Amazon EMR steps](#) in the Amazon EMR Management Guide.
3. Go to the AWS IAM console.
4. On the Roles page, choose **Create role**.
5. Choose **Custom trust policy**.
6. Enter information for the trust policy as shown in the example below, and edit it according to the project information you received in step 1.
 - Change `project-role-arn` to be the project role ARN you received from the Amazon SageMaker Unified Studio project member.
 - Change `project-id` to be the project ID you received from the Amazon SageMaker Unified Studio project member.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "project-role-arn"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

```
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "sts:ExternalId": "project-id"
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "project-role-arn"
            },
            "Action": [
                "sts:SetSourceIdentity"
            ],
            "Condition": {
                "StringLike": {
                    "sts:SourceIdentity": "${aws:PrincipalTag/datazone:userId}"
                }
            }
        }
    ]
}
```

7. Choose **Next**.
8. Under **Role name**, enter a name for the role.
9. (Optional) Enter a description for the role.
10. Choose **Create role**.

Attach permissions to the role

1. Select the role you have created in the AWS IAM console.
2. Choose **Add permissions > Create inline policy**.
3. Enter information as shown in the example below, and edit it according to the information for your Amazon EMR clusters that you want to grant access to.
 - Change the EMR cluster ARN to be the ARN for the cluster. You can find this on the cluster details page in the Amazon EMR console by selecting the cluster ID of the cluster that you want to share.

Note

You can use an asterisk instead of the Amazon EMR cluster ID if you want to grant access to all clusters instead of just one.

- Change the certificate path to the one defined in the Amazon EMR security configuration for that cluster in the Amazon EMR console. For more information, see [Specify a security configuration for an Amazon EMR cluster](#) in the Amazon EMR Management Guide.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "EmrAccess",  
            "Effect": "Allow",  
            "Action": [  
                "elasticmapreduce>ListInstances",  
                "elasticmapreduce>DescribeCluster",  
                "elasticmapreduce>GetClusterSessionCredentials" # Skip this for  
non-runtime role clusters  
            ],  
            "Resource": "arn:aws:elasticmapreduce:us-east-1:666777888999:cluster/j-  
AB1CDEFGHIJK" # EMR cluster ARN  
        },  
        {  
            "Sid": "EMRSelfSignedCertAccess",  
            "Effect": "Allow",  
            "Action": [  
                "s3:GetObject"  
            ],  
            "Resource": [  
                "arn:aws:s3:::666777888999-us-east-1-sam-dev/my-certs.zip" # Cert  
path defined in the EMR security configuration  
            ]  
        },  
        {  
            "Sid": "EMRSecurityConfigurationAccess",  
            "Effect": "Allow",  
            "Action": [  
        }
```

```
        "elasticmapreduce:DescribeSecurityConfiguration"
    ],
    "Resource": [
        "*"
    ]
}
```

4. Choose **Next**.
5. Under **Policy name**, enter a name for the policy.
6. Choose **Create policy**. You can then see the permissions policy listed on the page for the role you created in the IAM console.

Send information to project members

1. Copy the ARN of the EMR access role you created in the IAM console and send it to the Amazon SageMaker Unified Studio project member you want to grant access to.
2. Copy the Amazon EMR cluster ARN that you added to the permissions policy and send it to the Amazon SageMaker Unified Studio project member you want to grant access to.
3. From the Amazon EMR on EC2 cluster details page in the Amazon EMR console, copy the EC2 instance profile string and search for it on the Roles page in the IAM console to find the role that contains the Amazon EC2 instance profile ARN.
4. Select the name of the role that contains the instance profile ARN to open the role details page, then copy the ARN and send it to the Amazon SageMaker Unified Studio project member you want to grant access to.

After the Amazon EMR admin has completed these steps, project members are able to add a connection to the Amazon EMR on EC2 cluster as a compute resource in Amazon SageMaker Unified Studio.

Adding the Amazon EMR on EC2 compute resource

1. From inside the project management view in Amazon SageMaker Unified Studio, select **Compute** from the navigation bar.
2. On the Compute page, select the **Data processing** tab.

3. Choose **Add compute**, then choose **Connect to existing compute resources**.
4. In the **Add compute** modal, you can select the type of compute resource you would like to add to your project. Select **EMR on EC2 cluster**.
5. To add a connection to an existing Amazon EMR on EC2 cluster, you must have the correct permissions to access the Amazon EMR on EC2 cluster. You can select the **Copy project information** button to copy the data that the Amazon EMR admin will need to grant the data worker access. If you haven't already, send the project role ARN and the project ID to your admin.

 **Note**

The Amazon EMR admin will also need the project ID, which is the penultimate string in the project ARN. To view and copy the project ID, go to the **Project overview** page of your project.

6. After the account administrator has granted you access according to the prerequisite steps above, you can specify the ARNs associated with the cluster. You must fill in the **Access role ARN**, **EMR on EC2 cluster ARN**, **Compute name**, and the **Instance profile role ARN**.
7. Choose **Add compute**. Your Amazon EMR on EC2 instance is then added to your project.

After you have added a cluster to a project, you are able to see the cluster in the list on the **Data processing** tab in the Compute panel. You can then view the cluster details by selecting the cluster you want.

Using an Amazon EMR on EC2 cluster

After connecting to an Amazon EMR on EC2 cluster, you can begin using the cluster. To get started, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the compute connection. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. On the **Compute** page, choose the name of the compute you want to initialize. This takes you to a page with details about the cluster. Make a note of the name of the compute.
4. Choose **Actions > Open JupyterLab IDE**.

5. In the first cell, choose a connection type that you want to use from the dropdown list of connection types. Then choose the name of the compute from the dropdown list of compute options.
6. Choose the **Run** icon.

Your cluster is now initialized and configured to be a compute resource in your Amazon SageMaker Unified Studio project.

Monitoring Amazon EMR on EC2 clusters in Amazon SageMaker Unified Studio

You can monitor the performance of your Amazon EMR on EC2 clusters to ensure optimal resource use and efficient job execution. Information on metrics is automatically collected and sent to Amazon CloudWatch during operation of an Amazon EMR cluster.

You can see [CloudWatch metrics](#) for a specific cluster by selecting the cluster you're interested in from the list of clusters under the Cluster tab. Selecting a cluster will bring you to the Detail view for that cluster. After you've selected a cluster, select the **Monitoring** tab.

You will be able to see a grid view of the CloudWatch Metrics for the cluster you selected.

You can see information presented through different views by using the **Dashboard View** drop-down menu: Cluster Overview, Primary Node Group, Core Node Group, Task Node Group. You can also adjust the time range.

Terminating and removing an Amazon EMR on EC2 cluster

When you no longer need an Amazon EMR on EC2 cluster, the cluster can be terminated and removed.

To remove a cluster:

1. Login to the Amazon SageMaker Unified Studio and navigate to the **Data processing** tab of the Compute section. Select the name of the compute instance you would like to remove.
2. On the compute details page, select the **Terminate and remove** option.
3. A dialog box will appear asking you to confirm that you want to terminate and remove the instance of compute, which in this case is your Amazon EMR on EC2 cluster. Confirm that you want to remove the compute, by typing "confirm" in the text box.

4. Choose **Terminate and remove compute** to begin termination and removal.
5. After a few minutes, your cluster should have been removed.

Spark History Server

You can use the live Spark UI in a notebook session to view details such as tasks, executors and logs about Spark jobs.

You can explore the Spark History Server for a cluster at any time. To do this, select your cluster from the list of all clusters assigned to a project, which brings up the Detail view for the cluster. On the Detail page view, select the **Applications** tab and choose the '**Spark History Server**' link.

EMR Serverless compute connections in Amazon SageMaker Unified Studio

In addition to Amazon EMR on EC2 clusters, you can also create and delete EMR Serverless applications.

Adding a new EMR Serverless application

As a data worker, you can make use of EMR Serverless applications by adding them to a project in the Amazon SageMaker Unified Studio Studio. Within a project, you can use both existing and new applications. You can use existing applications at any time. However, in order to create a new EMR Serverless application, the admin must enable blueprints.

After your admin has enabled blueprints:

1. From inside the project management view, select **Compute** from the navigation bar.
2. In the Compute panel, select the **Data processing** tab.
3. To add an instance of an Amazon EMR Serverless, select the **Add compute** dropdown menu and then choose **New compute**.
4. In the **Add compute** modal, you can select the type of compute you would like to add to your project. Select **EMR Serverless**.
5. The **Add compute** dialog box allows you to specify the name of the EMR Serverless application, provide a description, and choose a release of EMR Serverless that you want your application to use.

6. Choose the permission mode option that supports the data you will be using with the compute resource.
 - Select **project.spark.fineGrained** for data managed using fine-grained access, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option configures your compute to work with data asset subscriptions from Amazon SageMaker catalog.
 - Select **project.spark.compatibility** to configure permission mode to be compatible with data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option configures your compute to work with data assets from AWS and from external systems that you connect to from your project.
7. After configuring these settings, select **Add compute**. After a short time, your serverless application running EMR Serverless should be added to your project.

Configuring permission mode for EMR Serverless in Amazon SageMaker Unified Studio

Permission mode is a configuration available to Spark compute resources such as Glue ETL or EMR Serverless. It configures Spark to access different types of data based on the permissions configured for that data. There are two configuration options for permission mode:

- Compatibility mode. This is a configuration for data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option enables your compute to work with data assets from AWS and from external systems.
- Fine-grained mode. This is a configuration for data managed using fine-grained access controls, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option enables your Glue ETL to work with data asset subscriptions from Amazon SageMaker catalog.

You configure permission mode in EMR Serverless computes in Amazon SageMaker Unified Studio when you add a new EMR Serverless compute resource in your project. For more information, see [the section called “Adding a new EMR Serverless application”](#).

Note

You cannot modify the permission mode after the EMR Serverless compute resource is created. Instead, you can create another EMR Serverless compute resource with a different permission mode.

Deleting applications

When you no longer need an EMR Serverless application, the application can be deleted.

To delete an application:

1. Login to the Amazon SageMaker Unified Studio studio and navigate to the Serverless tab of the Compute section. Select the name of the compute instance you would like to remove.
2. On the compute details page, select the **Delete** option.
3. A dialog box will appear asking you to confirm that you want to delete the application, which in this case is your EMR Serverless application. Confirm that you want to remove the compute by typing "confirm" in the text box.
4. Choose **Delete application** to begin termination and removal.
5. After a short time, your application should be removed.

Glue ETL in Amazon SageMaker Unified Studio

AWS Glue ETL compute resources power Visual ETL flows in your Amazon SageMaker Unified Studio project. You can use Glue ETL serverless compute resources to run Visual ETL flows and JupyterLab notebooks without managing underlying infrastructure. This is especially useful for analytics, machine learning, and application development.

You can view information about your AWS Glue ETL compute resources on the **Data processing** tab of the **Compute** page in your project. These resources are used when you create and run Visual ETL flows in Amazon SageMaker Unified Studio.

By default, when you create a project in Amazon SageMaker Unified Studio two Glue ETL compute connections are created. The Glue ETL connection with permission mode set to compatibility is called `project.spark.compatibility`, and the Glue ETL connection with permission mode set

to fine-grained is called `project.spark.fineGrained`. You can choose which compute option to use when you use tools such as Visual ETL and JupyterLab in Amazon SageMaker Unified Studio. For more information about compatibility and fine-grained permission modes, see [the section called “Configuring permission mode”](#).

 **Note**

Amazon SageMaker Unified Studio automatically creates Glue ETL compute resources during project creation. You cannot create, edit, or delete these instances. By default, Glue ETL uses AWS Glue 5.0 with G.1X (1 Data Processing Unit / Hour) worker types.

Configuring permission mode for Glue ETL in Amazon SageMaker Unified Studio

Permission mode is a configuration available to Spark compute resources such as Glue ETL or EMR Serverless. It configures Spark to access different types of data based on the permissions configured for that data. There are two configuration options for permission mode:

- Compatibility mode. This is a configuration for data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option enables your compute to work with data assets from AWS and from external systems.
- Fine-grained mode. This is a configuration for data managed using fine-grained access controls, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option enables your Glue ETL to work with data asset subscriptions from Amazon SageMaker catalog.

To configure permission mode for Glue ETL in Amazon SageMaker Unified Studio, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to a project.
3. Navigate to the Visual ETL tool by using the dropdown **Build** menu and selecting **Visual ETL flows**.
4. Navigate to a flow by creating one or selecting the flow from the list.

5. From the dropdown menu next to the **Run** button, choose a compute connection type that aligns with your data access preference.
 - Select **project.spark.fineGrained** to configure permission mode to support fine-grained access control. Choosing this option configures your Visual ETL flow to work with data asset subscriptions from Amazon SageMaker catalog.
 - Select **project.spark.compatibility** to configure permission mode to be compatible with general access control. Choosing this option configures your Visual ETL flow to work with data assets that you connect to from your project.

You can then run the Visual ETL flow with data that aligns with your selected compute connection.

Limitations of fine-grained permission mode

Permission mode is a configuration available to Spark compute resources such as Glue ETL or EMR Serverless. It configures Spark to access different types of data based on the permissions configured for that data. There are two configuration options for permission mode:

- Compatibility mode. This is a configuration for data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option enables your compute to work with data assets from AWS and from external systems.
- Fine-grained mode. This is a configuration for data managed using fine-grained access controls, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option enables your Glue ETL to work with data asset subscriptions from Amazon SageMaker catalog.

Consider the following considerations and limitations when you use fine-grained mode.

- Fine-grained mode supports fine-grained access control via AWS Lake Formation only for Apache Hive and Apache Iceberg tables. Apache Hive formats include Parquet, ORC, and CSV.
- When fine-grained mode is enabled, a minimum of four workers are required because it requires one system driver, system executors, one user driver, and optionally user executors (required if you use UDFs or spark.createDataFrame).
- Fine-grained mode supports cross-account table queries shared through resource links. The resource link needs to be named identically to the source account's resource.
- The following components aren't supported:

- Resilient distributed datasets (RDD)
- Spark streaming
- Write with AWS Lake Formation granted permissions
- Access control for nested columns
- Access data stored on Amazon Redshift Managed Storage (RMS), including through Amazon SageMaker Lakehouse.
- Fine-grained mode blocks functionalities that might undermine the complete isolation of the system driver, including the following:
 - UDTs, HiveUDFs, and any user-defined function that involves custom classes
 - Custom data sources
 - Supply of additional JARs for Spark extension, connector, or metastore
 - ANALYZE TABLE command
- To enforce access controls, EXPLAIN PLAN and DDL operations such as DESCRIBE TABLE don't expose restricted information.
- Fine-Grained mode restricts access to system driver Spark logs on Lake Formation-enabled applications. Since the system driver runs with more access, events and logs that the system driver generates can include sensitive information. To prevent unauthorized users or code from accessing this sensitive data, access to system driver logs is disabled. For troubleshooting, contact AWS support.
- The following are considerations and limitations when using Apache Iceberg:
 - You can only use Apache Iceberg with session catalog and not arbitrarily named catalogs.
 - Iceberg tables that are registered in AWS Lake Formation only support the metadata tables `history`, `metadata_log_entries`, `snapshots`, `files`, `manifests`, and `refs`. AWS Glue hides the columns that might have sensitive data, such as `partitions`, `path`, and `summaries`. This limitation doesn't apply to Iceberg tables that aren't registered in AWS Lake Formation.
 - Tables that you don't register in AWS Lake Formation support all Iceberg stored procedures except for the `register_table` and `migrate` procedures, which aren't supported for any tables.

Visual ETL

Data engineers, analysts, and scientists use visual ETL features to create extract, transform, and load (ETL) flows using an intuitive visual interface. With visual ETL, analytics users can discover, prepare, move, and integrate data from multiple sources. This simplifies the process of data manipulation and integration so that you can prepare data for analysis and reporting.

Visual ETL in Amazon SageMaker Unified Studio provides a drag-and-drop interface for building ETL flows and authoring flows with Amazon Q. You can connect to data sources, apply transformations, and define target destinations without writing complex code.

You can use Visual ETL to implement solutions such as:

- Data integration from multiple sources
- Data cleansing and normalization
- Creating data warehouses or data lakes
- Preparing data for machine learning models
- Automating regular data processing tasks

Authoring flows with Visual ETL utilizes AWS Glue interactive sessions Version 5.0.

Key features

Visual ETL offers several capabilities to streamline your data workflows:

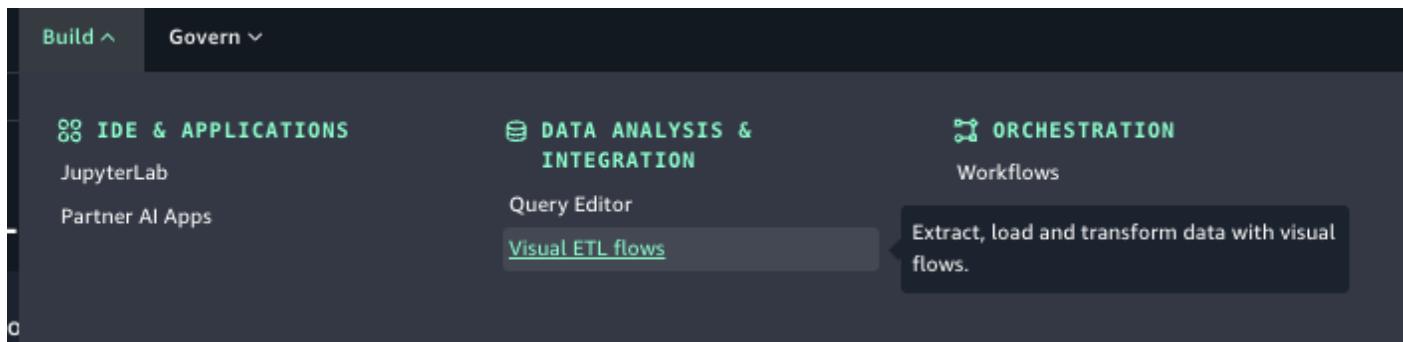
1. Drag-and-drop interface: Create Visual ETL flows by dragging and connecting components on a canvas.
2. Wide range of data connectors: Connect to various data sources and destinations, including databases, file systems, cloud storage, and APIs.
3. Extensive transformation library: Apply a variety of pre-built transformations to your data, such as filtering, aggregation, joining, and data type conversions.
4. Custom transformations: Create and save custom transformations using SQL or Python for reuse in multiple flows.
5. Data preview: Visualize your data at each step of the authoring process to ensure accuracy and data quality.

6. View scripts: View the code generated and choose to convert the flow to a notebook and continue authoring with code.
7. Code and compute configuration: Use a configuration panel to add code libraries and adjust the compute settings.

Creating a Visual ETL flow

To create a flow using Visual ETL in Amazon SageMaker Unified Studio:

1. Log in to Amazon SageMaker Unified Studio and select a project.
2. Navigate to the Visual ETL tool using the dropdown "Build" menu, selecting "Visual ETL flows".

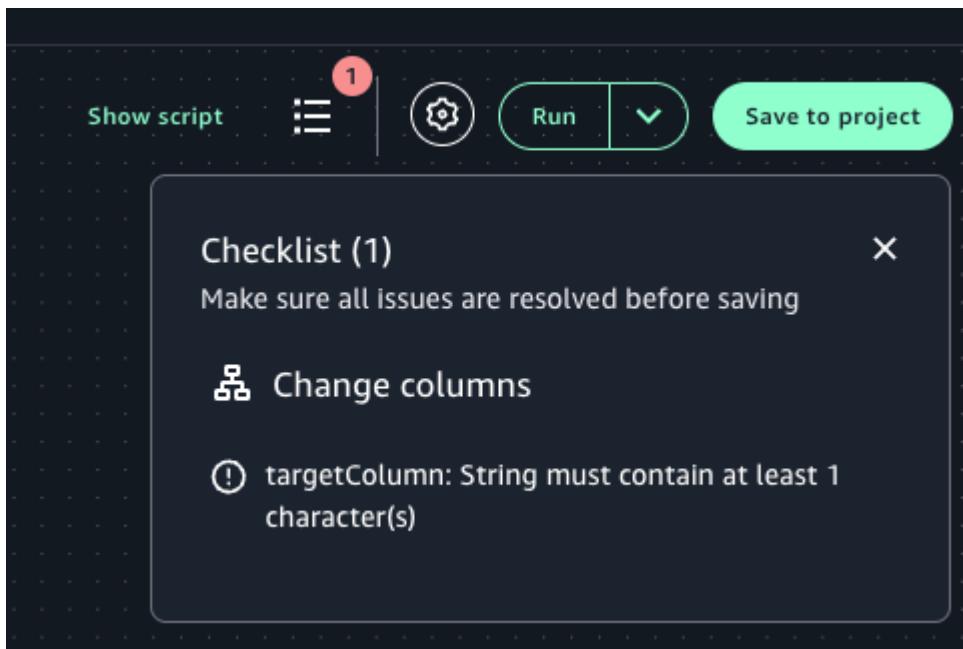


3. Click "Create visual ETL flow" to open the Visual ETL editor.

If this is your first time using Visual ETL flows in Amazon SageMaker Unified Studio, you are asked to choose a default compute permission mode option based on your data access preference. For more information, see [the section called "Configuring permission mode"](#).

4. Give the flow a name when you begin authoring the flow.
5. From the dropdown menu next to the Run button, choose the compute permission mode option that supports the data you will be using in the flow.
 - Select **project.spark.fineGrained** for data managed using fine-grained access, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option configures your compute to work with data asset subscriptions from Amazon SageMaker catalog.
 - Select **project.spark.compatibility** to configure permission mode to be compatible with data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option configures your compute to work with data assets from AWS and from external systems that you connect to from your project.

6. Select the "Add nodes" button and select a node, choosing your node from one of the three tabs: "Data sources", "Transforms", or "Data targets".
 7. Drag a source component onto the canvas.
 8. Configure the component by clicking on the node and editing the configurations, to connect to your data source.
 9. Add transformation components as needed, connecting them in the desired order.
10. Drag a data target onto the canvas and configure it to specify where the processed data should be stored.
11. Connect the components to create a complete flow.



12. Click the "Checklist" button to check for any configuration errors.
13. To make the flow accessible for all project members to view and edit, select "Save to project".
14. Select "Run" to execute it immediately or run it on a schedule with the instructions at [the section called "Scheduling and running visual flows"](#).

Supported connectors for Visual ETL

Visual ETL in Amazon SageMaker Unified Studio supports the connectors listed at [the section called "Supported data sources"](#), in addition to Amazon S3 and Amazon SageMaker AI Lakehouse.

For instructions on how to add a new connection, see [the section called "Adding data sources"](#).

Supported transforms for Visual ETL

The following section contains information on the supported transforms for Visual ETL in Amazon SageMaker Unified Studio :

Topics

- [Add current timestamp transform](#)
- [Aggregate transform](#)
- [Array to columns transform](#)
- [Autobalance processing transform](#)
- [Change columns transform](#)
- [Concatenate columns transform](#)
- [Custom code transform](#)
- [Derived column transform](#)
- [Drop columns transform](#)
- [Drop duplicates transform](#)
- [Drop nulls transform](#)
- [Explode array or map into rows transform](#)
- [Extract JSON path transform](#)
- [Fill nulls transform](#)
- [Filter transform](#)
- [Flatten transform](#)
- [Format timestamp transform](#)
- [Identifier transform](#)
- [Intersect transform](#)
- [Join transform](#)
- [Limit tranform](#)
- [Lookup transform](#)
- [Parse JSON column transform](#)
- [Pivot rows into columns transform](#)

- [Regex extract transform](#)
- [Rename columns transform](#)
- [Select columns transform](#)
- [Split string transform](#)
- [SQL query transform](#)
- [To timestamp transform](#)
- [Union transform](#)
- [Unpivot columns into rows transform](#)
- [UUID transform](#)

Add current timestamp transform

The Add Current Timestamp transform allows you to mark the rows with the time on which the data was processed. This is useful for auditing purposes or to track latency in the data pipeline. You can add this new column as a timestamp data type or a formatted string.

To add an Add Current Timestamp transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Add Current Timestamp**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Timestamp column name**, enter a custom name for the new column.
7. (Optional) under **Timestamp format**, enter a format if you would prefer the column to be a formatted date string.

Aggregate transform

You may use the Aggregate transform to perform summary calculations on selected fields.

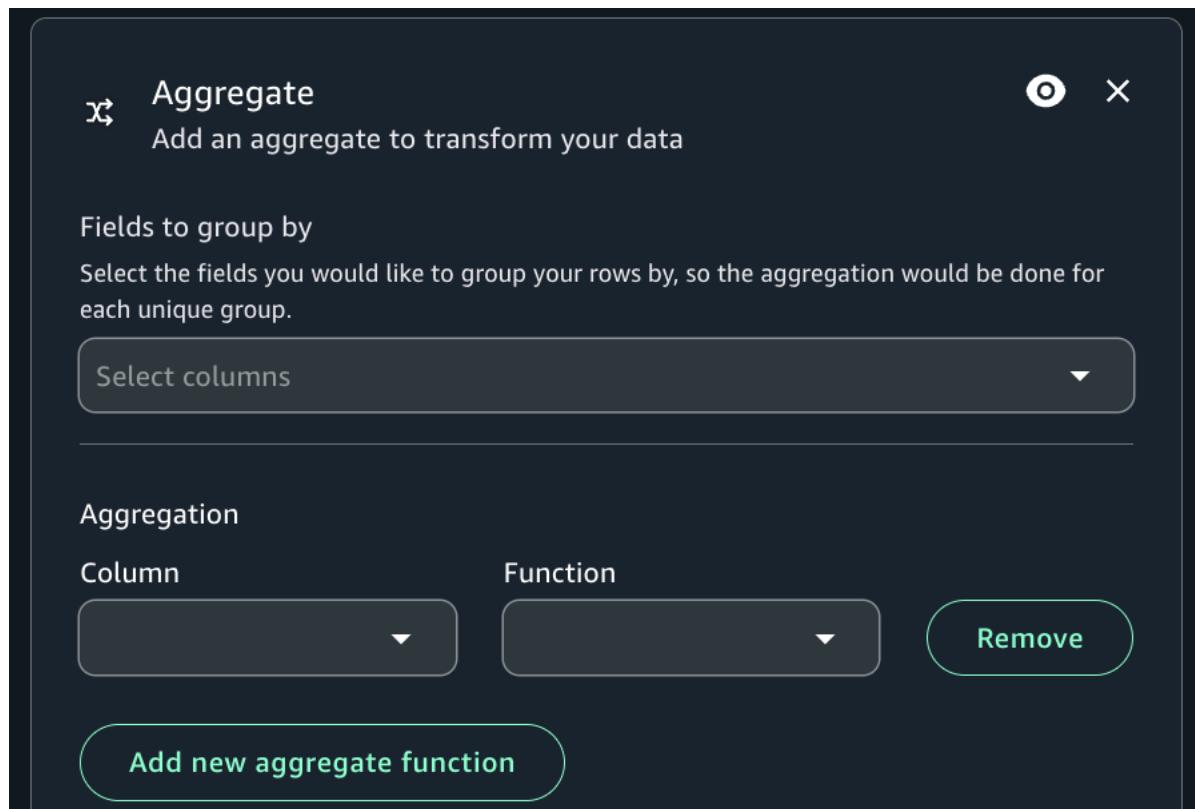
To use the Aggregate transform

1. Add the Aggregate node to the visual flow diagram.

2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. On the Node properties view, choose the "fields to group by", selecting the drop-down field (optional). You can select more than one field at a time or search for a field name by typing in the search bar.

When fields are selected, the name and datatype are shown. To remove a field, click 'X' on the field.

4. Choose Aggregate another column. It is required to select at least one field.
5. Choose a field in the Field to aggregate drop-down.
6. Choose the aggregation function to apply to the chosen field:
 - avg - calculates the average
 - count - calculates the number of non-null values
 - max - returns the highest value that satisfies the 'group by' criteria
 - min - returns the lowest value that satisfies the 'group by' criteria
 - sum - the sum of all values in the group



Array to columns transform

The Array To Columns transform allows you extract some or all the elements of a column of type array into new columns. The transform will fill the new columns as much as possible if the array has enough values to extract, optionally taking the elements in the positions specified.

For instance, if you have an array column “subnet”, which was the result of applying the “Split String” transform on a ip v4 subnet, you can extract the first and forth positions into new columns “first_octect” and “forth_octect”. The output of the transform in this example would be (notice the last two rows have shorter arrays than expected):

subnet	first_octect	fourth_octect
[54, 240, 197, 238]	54	238
[192, 168, 0, 1]	192	1
[192, 168]	192	
[]		

To add an Array to Columns transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Array to Columns**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Array type column**, choose the column of type array from which the new columns are extracted.
7. Under **Output columns**, enter names for the output columns.
8. (Optional) Under **Array indexes to use**, enter numbers to indicate which columns to include.

Autobalance processing transform

Use this transform to repartition data to optimize future cluster resource usage. This transform is particularly useful for uneven datasets.

To add an Autobalance Processing transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Autobalance Processing**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Number of partitions**, input a number of partitions to randomly distribute the data into. Or, switch the toggle to off to use the number of cores as the partition number.
7. (Optional) Under **Repartition columns**, identify columns that you want data of the same value to be assigned to the same partition in.

Change columns transform

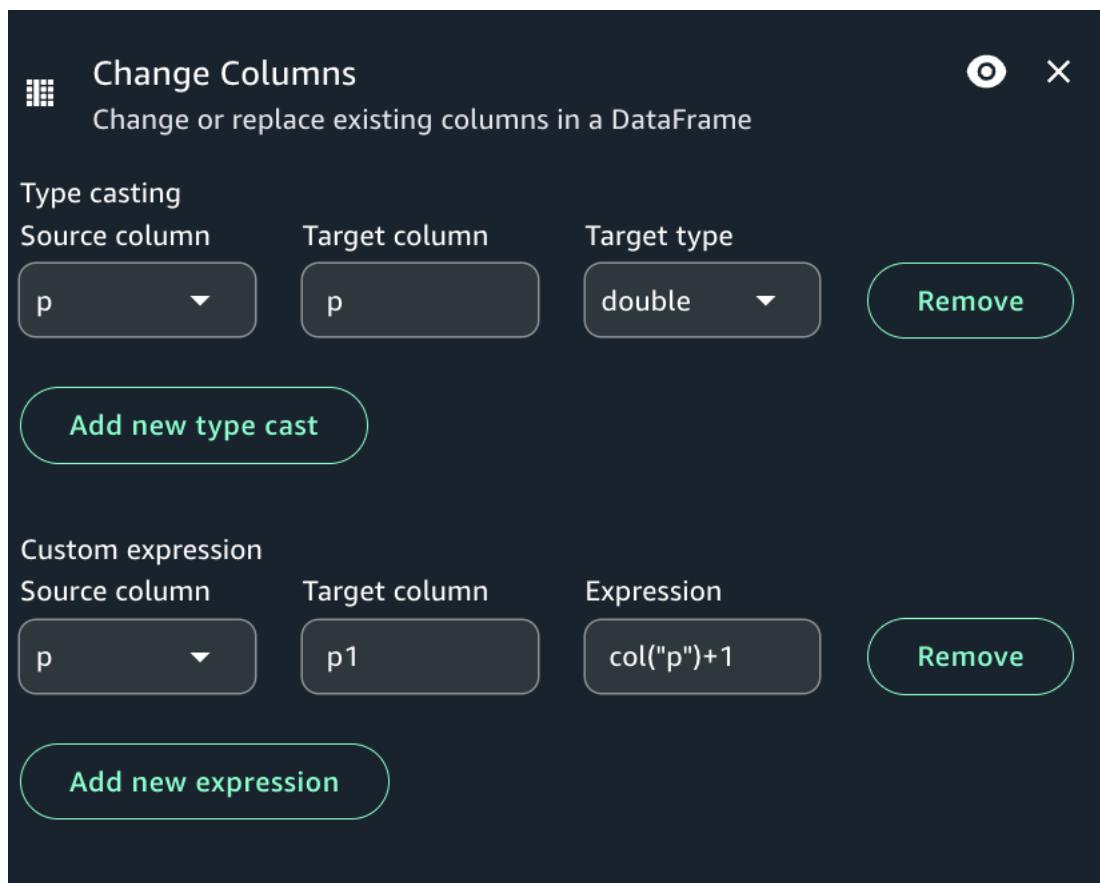
A Change columns transform remaps the source data property keys into the desired configured for the target data. In a Change Columns transform node, you can:

1. Change the name of multiple columns.
2. Change the data type of the columns, if the new data type is supported and there is a transformation path between the two data types.
3. Use a custom expression to change the values of selected columns.

To add a Change Schema node to your flow diagram

1. Open the menu and then choose Change Columns to add a new transform to your flow diagram, if needed.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema by clicking on the node, then:
 1. To rename a column, enter the new name in the Target column field.

2. To change the data type for a selected column, select "Add new type cast", then choose the source column, target column, and new data type from the Target type list.
3. To change the data values for a selected column, add a custom expression.
4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.



Concatenate columns transform

The Concatenate transform allows you to build a new string column using the values of other columns with an optional spacer. For example, if we define a concatenated column “date” as the concatenation of “year”, “month” and “day” (in that order) with “-” as the spacer, we would get:

day	month	year	date
01	01	2020	2020-01-01

day	month	year	date
02	01	2020	2020-01-02
03	01	2020	2020-01-02
04	01	2020	2020-01-02

To add a Concatenate Columns transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Concatenate Columns**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Concatenated column**, enter the name of a new column that will be generated.
7. Under **Columns**, select the input columns.
8. (Optional) Under **Spacer**, enter a string to place between concatenated fields.
9. (Optional) Under **Null value**, enter the string to use when a column value is null.

Custom code transform

If you need to perform more complicated transformations on your data, or want to add data property keys to the dataset, you can add a Custom code transform to your flow diagram. The Custom code node allows you to enter a script that performs the transformation.

To add a custom code node to your flow diagram

1. Open the Resource panel and then choose Custom Code to add a custom transform to your flow diagram.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Enter desired code changes.

The following examples show the format of the code to enter in the code box:

```
def FilterPopulationAbove1000(input_df):
    df = input_df
    df = df[df['population'] > 1000]
    return df
```

```
1  # Imports
2
3
4  def FilterPopulationAbove1000(input_df):
5      df = input_df
6      df = df[df['population'] > 1000]
7      return df
```

Python Ln 0, Col 0 Errors: 0

Derived column transform

The Derived Column transform allows you to define a new column based on a math formula or SQL expression in which you can use other columns in the data, as well as constants and literals. For instance, to derive a “percentage” column from the columns “success” and “count”, you can enter the SQL expression: “`success * 100 / count || '%'`”.

Example result:

success	count	percentage
14	100	14%
6	20	3%
3	40	7.5%

To add a Derived Column transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Derived Column**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Name of derived column**, enter the name of a new column that will be generated.
7. Under **Column expression**, enter a SQL expression to define the new column based on existing columns.

Drop columns transform

You can create a subset of data property keys from the dataset using the Drop columns transform. You indicate which data property keys you want to remove from the dataset and the rest of the keys are retained.

To add a Drop columns transform node to your flow diagram

1. Open the Resource panel and then choose Drop columns to add a new transform to your diagram.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Click on the node and view the Node properties panel.
4. Choose the data property keys from the "Field" column to drop the column from the data source.

5. (Optional) After configuring the node properties and transform properties, you can preview using the Data preview tab in flow diagram.

The screenshot shows the 'Drop Columns' configuration dialog. It lists several fields with their data types. The 'latitude' field is checked, while others like 'longitude', 'housing_me...', 'total_rooms', and 'total_bedro...' are not checked. The dialog includes a header with a close button, a table with columns for Field and Data type, and a footer with pagination controls.

Drop duplicates transform

The Drop duplicates transform removes rows from your data source by giving you two options. You can choose to remove the duplicate row that are completely the same, or you can choose to choose the fields to match and remove only those rows based on your chosen fields.

For example, in this data set, you have duplicate rows where all the values in some of the rows are exactly the same as another row, and some of the values in rows are the same or different.

Example Data Set

Row	Name	Email	Age	State	Note
1	Joy	joy@gmail	33	NY	

Row	Name	Email	Age	State	Note
2	Tim	tim@gmail	45	OH	
3	Rose	rose@gmail	23	NJ	
4	Tim	tim@gmail	42	OH	
5	Rose	rose@gmail	23	NJ	
6	Tim	tim@gmail	42	OH	this is a duplicate row and matches completely on all values as row #4
7	Rose	rose@gmail	23	NJ	This is a duplicate row and matches completely on all values as row #5

If you choose to match entire rows, rows 6 and 7 will be removed from the data set. The data set is now:

Data Set After Matching Entire Rows

Row	Name	Email	Age	State
1	Joy	joy@gmail	33	NY
2	Tim	tim@gmail	45	OH
3	Rose	rose@gmail	23	NJ
4	Tim	tim@gmail	42	OH

Row	Name	Email	Age	State
5	Rose	rose@gmail	23	NJ

If you chose to specify keys, you can choose to remove rows that match on 'name' and 'email'. This gives you finer control of what is a 'duplicate row' for your data set. By specifying 'name' and 'email', the data set is now:

Data Set After Specifying Keys

Row	Name	Email	Age	State
1	Joy	joy@gmail	33	NY
2	Tim	tim@gmail	45	OH
3	Rose	rose@gmail	23	NJ

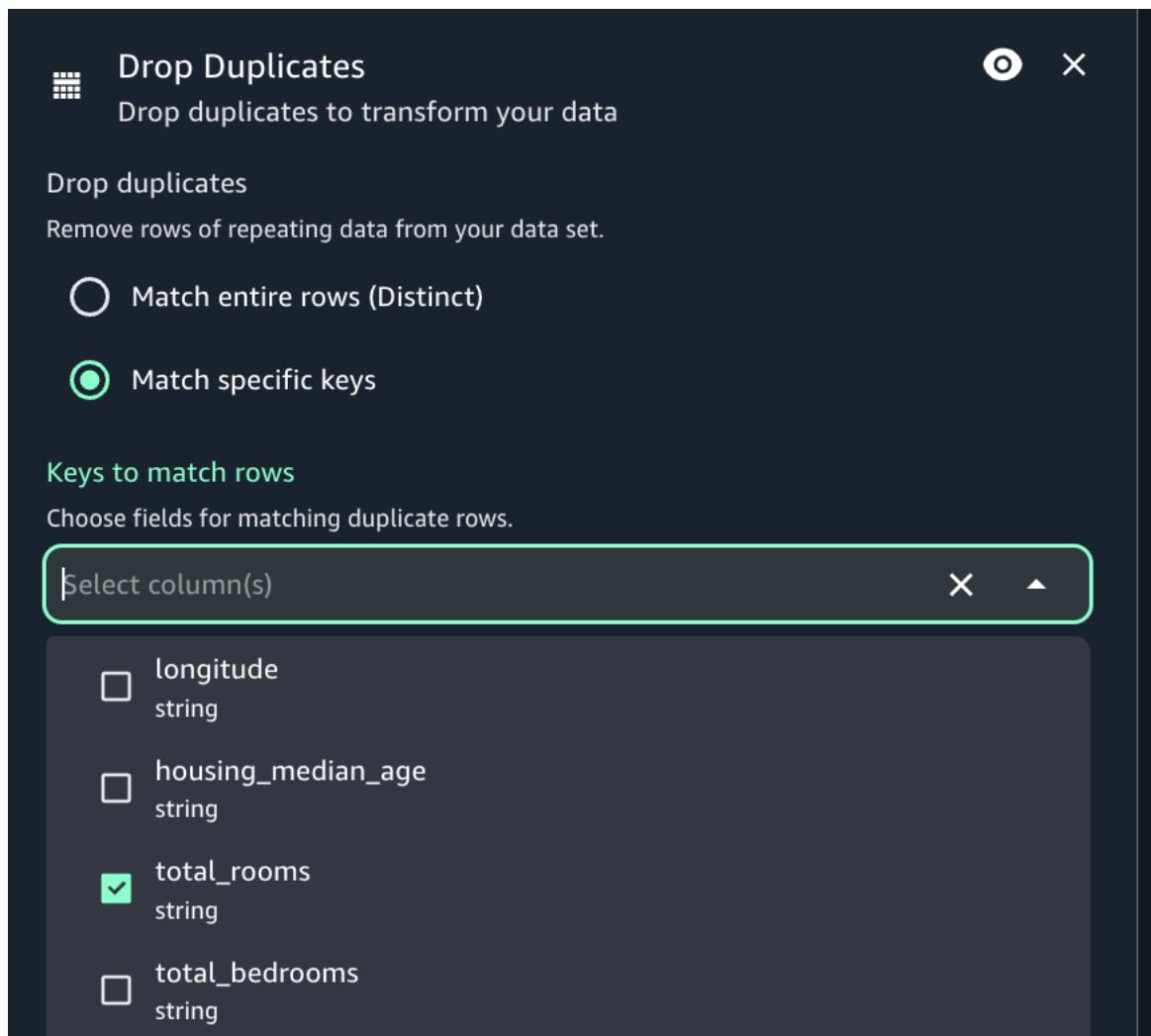
Some things to keep in mind:

- In order for rows to be recognized as a duplicate, values are case sensitive. all values in rows need to have the same casing - this applies to either option you choose (Match entire rows or Specify keys).
- All values are read in as strings.
- The Drop duplicates transform utilizes the Spark dropDuplicates command.
- When using the Drop duplicates transform, the first row is kept and other rows are dropped.
- The Drop duplicates transform does not change the schema of the dataframe.

To add a Drop duplicates transform node to your flow diagram

1. Open the Resource panel and then choose Drop duplicates to add a new transform to your diagram.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Click on the node and view the Node properties panel.
4. Choose if you prefer to drop duplicates by matching entire rows or specific keys.

5. (Optional) After configuring the node properties and transform properties, you can preview using the Data preview tab in flow diagram.



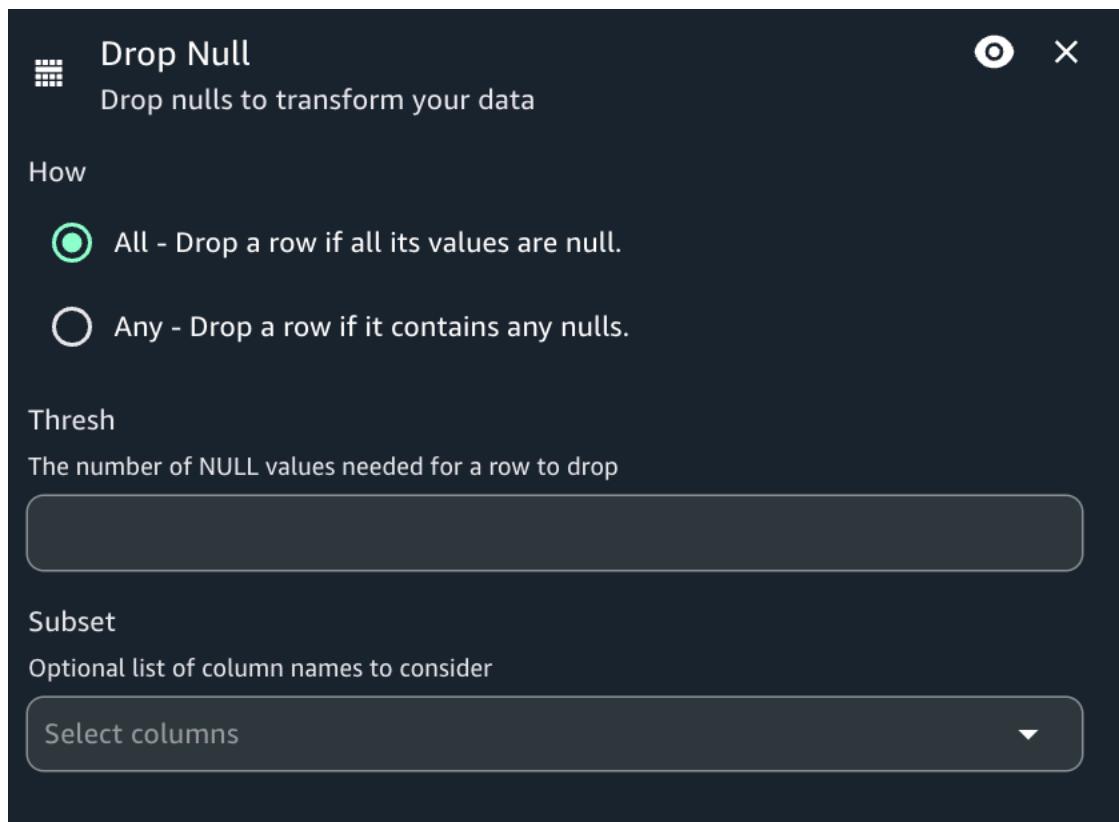
Drop nulls transform

Use the Drop nulls transform to remove fields from the dataset if all values in the field are 'null'. By default, Visual ETL will recognize null objects, but some values such as empty strings, strings that are "null", -1 integers or other placeholders such as zeros, are not automatically recognized as nulls.

To use the Drop nulls

1. Add a Drop nulls node to the diagram, if needed.
2. Set a number as threshold value, if the rows(s) need to be drop only if a certain number of nulls is present.

3. Choose if you prefer to drop nulls for all columns or only check null values for specific columns. If you choose the subset option, select the desired column names from the drop down list.



Explode array or map into rows transform

You can use the Explode transform to extract values from a nested structure into individual rows that are easier to manipulate. In the case of an array, the transform will generate a row for each value of the array, replicating the values for the other columns in the row. In the case of a map, the transform will generate a row for each entry with the key and value as columns plus any other columns in the row.

For example, if we have this dataset which has a "category" array column with multiple values.

product_id	category
1	[sports, winter]
2	[garden, tools]

product_id	category
3	[videos]
4	[games, electronics, social]
5	[]

If you explode the 'category' column into a column with the same name, you will override the column. You can select that you want NULLs included to get the following (ordered for illustration purposes):

1	sports
1	winter
2	garden
2	tools
3	videos
4	games
4	electronics
4	social
5	

To add an Explode Array Or Map Into Rows transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Explode Array or Map Into Rows**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.

6. Under **Column to explode**, enter the name of an existing column that will be exploded.
7. Under **New column name**, enter the name of a new column that will be generated.
8. (Optional) Under **Values column**, if you are exploding a dictionary, specify a name for a column to contain the values.
9. Use the toggle to indicate whether or not to include null values. By default, if the column to explode is NULL or has an empty structure, it will be omitted on the exploded dataset.

Extract JSON path transform

Use this transform to extract new columns from a string column with JSON data.

To add an Extract JSON path transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Extract JSON Path**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **JSON column**, select a column from the data source.
8. Under **JSON paths**, enter a JSON path or multiple paths separated by commas to specify the location to extract the data from.
9. Under **Output columns**, enter the names of the columns to create from each of the JSON paths.
10. (Optional) If desired, use the toggle to drop the JSON column.

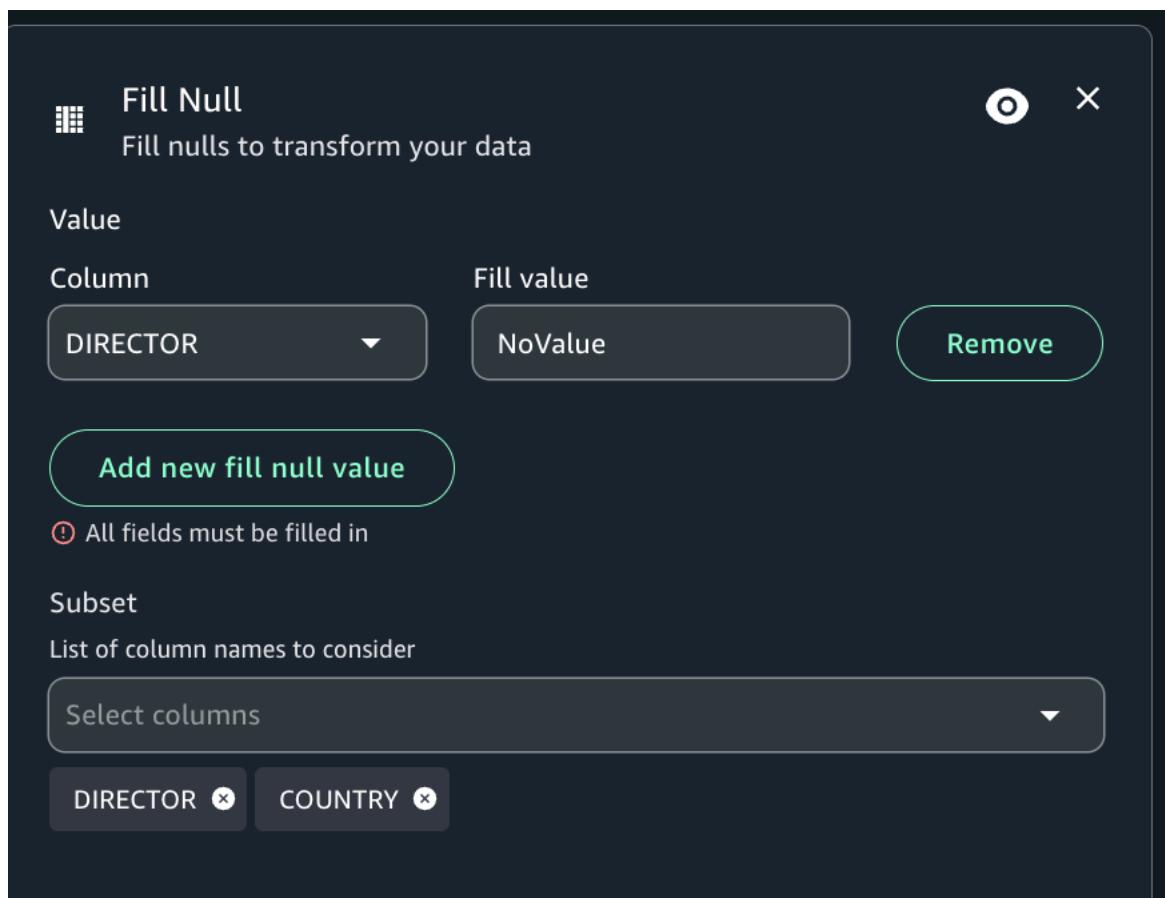
Fill nulls transform

Fill nulls allows you to fill in null value in a column with a chosen fill value. Select a column with nulls you want to fill and provide a fill value. You can use multiple fill conditions at one time. Select a subset of the data to consider by using a dropdown menu to select columns.

To add a Fill nulls node to your flow diagram

1. Open the menu and then choose Fill nulls to add a new transform to your flow diagram.

2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema:
 1. To create a subset, use the dropdown menu and select a column or columns to consider.
 2. To specify a value to fill nulls with, select "Add new fill null value", then choose a column to fill and provide a Fill value.
4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.

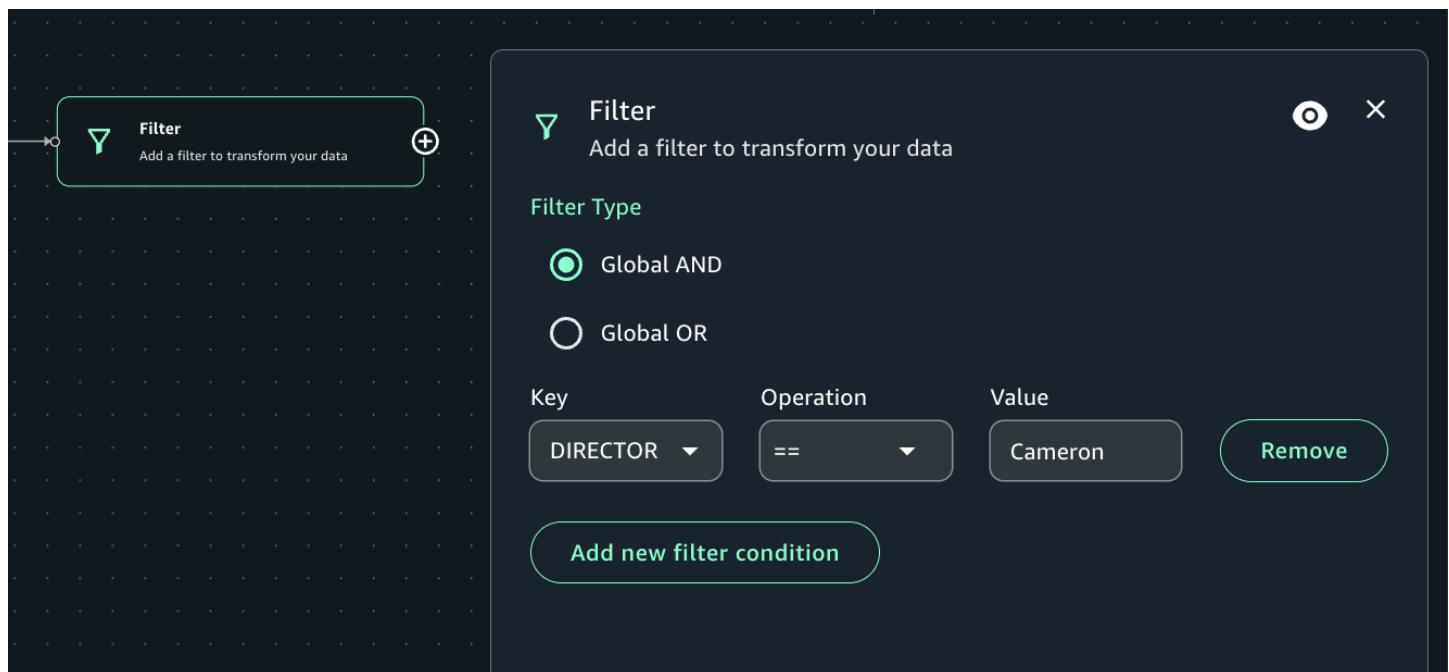


Filter transform

Use the Filter transform to create a new dataset by filtering records from the input dataset. Rows that don't satisfy the filter condition are removed from the output. You can select from two filter types: Global AND or Global OR. You must select a column name to serve as the key, a comparison operation, and provide a value to filter on.

To add a Filter node to your flow diagram

1. Open the menu and then choose Filter to add a new transform to your flow diagram.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema:
 1. Select "Add new filter condition".
 2. Choose a filter type: "Global AND" or "Global or".
 3. Select a Key column to filter.
 4. Select a comparison operation.
 5. Type in a value to compare in the "Value" box.
4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.



For example, if the data has a field of type Struct named “phone_numbers”, which among other fields has one of type “Struct” named “home_phone” with two fields: “country_code” and “number”. After they are flattened, these two fields will become top level fields named “phone_numbers.home_phone.country_code” and “phone_numbers.home_phone.number” respectively.

To add a Flatten transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Flatten**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Max levels to flatten**, enter the a maximum limit.
7. Under **Separator**, enter a character to use in the new column names to separate the different levels.

Format timestamp transform

Use this transform to convert a timestamp colum into a formatted string.

To add a Format Timestamp transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Format Timestamp**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Column**, choose the column that contains the timestamps.
7. Under **Timestamp format**, input the format you want to use for the strings.
8. (Optional) Enter a name for the new formatted column.

Identifier transform

Use this transform to assign a numeric identifier for each row in the dataset.

To add an Identifier transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Identifier**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **Column name**, enter a name for the new column. This column is named id by default.
8. (Optional) If you want to make the ID unique among the job runs, switch the toggle to on to include a timestamp and extend the column type from long to decimal.

Intersect transform

Use this transform to select only the common rows in two datasets. Duplicates are removed by default, and you can choose to keep them in by checking the **Intersect all** checkbox.

To add an Intersect transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Intersect**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. (Optional) If you want to intersect duplicated rows as well, check the **Intersect all** checkbox.

Join transform

The Join transform allows you to combine two datasets into one. You specify the key names in the schema of each dataset to compare. The output frame contains rows where keys meet the join condition. The rows in each dataset that meet the join condition are combined into a single row in the output from that contains all the columns found in either dataset. You can select from one of the following join types: Inner, Left, Right, Full, Cross, Semi, and Anti.

To add a Join node to your flow diagram

1. Open the menu and then choose Join to add a new transform to your flow diagram, if needed.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Optional) Ensure two data sources are connected to the Join node.
4. Modify the input schema:
 1. Select a join type from the "Join type" dropdown menu.(Optional).
 2. Select a column for the "Left data source" using the dropdown menu.
 3. Select a column for the "Right data source" using the dropdown menu.
5. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.



Note

If you have columns using the same names in your data sources, the join will result in a COLUMN_ALREADY_EXISTS error. To avoid this error, either: (1) use Rename columns before the join for one of your data sources or (2) use Drop Columns after the join to remove both duplicated columns.

Limit transform

Use this transform to limit the number of rows in a dataset.

To add a Limit transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Limit**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Number of records**, enter the maximum number of rows you want to see or process from the dataset.

Lookup transform

Use this transform to add matching columns by looking them up on another catalog table.

To add a Lookup transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Lookup**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **Catalog**, select the catalog to use for lookup.
8. Under **Database**, select a database.
9. Under **Table**, select a table.
10. Under **Lookup key columns to match**, enter columns in the lookup table, separated by commas.
11. Under **Lookup columns to take**, enter columns in the lookup table to add to the data when a match is found.

Parse JSON column transform

Use this transform to parse a string column that contains JSON data.

To add a Parse JSON Column transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Parse JSON Column**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **JSON column**, choose the column that contains JSON data
8. Enter a number to use as a ratio of samples to use for inferring schema. This must be a value between 0.1 and 1, with 1 meaning all samples are used. If your JSON strings are consistent, you can use a lower value to speed up the inference.
9. (Optional) Enter a name for the new column.

Pivot rows into columns transform

Use this transform to aggregate the values of a numeric column into categories by using the values of the columns configured.

To add a Pivot Rows Into Columns transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Pivot Rows Into Columns**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **Aggregation column**, select a numeric column to use.
8. Under **Aggregation**, select a Spark function to apply to the aggregation column.
9. Under **Columns to convert**, select columns whose values will become new columns.

Regex extract transform

Use this transform to extract new columns from a string column with JSON data.

To add a Regex Extract transform:

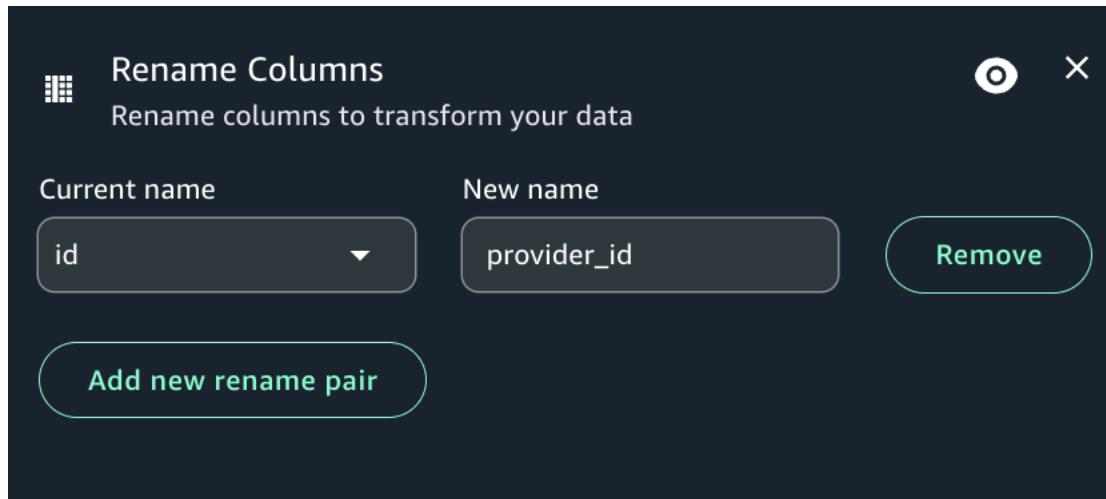
1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Regex Extract**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Column to extract from**, choose a column from the data source you connected to the transform.
7. Under **Regular expression**, enter a regular expression to apply on the column. If there are multiple columns, the expression must have a number of groups equal to the number of columns.
8. Under **Extracted column**, enter the name of the column to contain the extracted regex.

Rename columns transform

You can use the Rename columns transform to change the name of multiple columns in the dataset. You select a column that you want to rename from a list of available columns and provide the string that you want to update the column name to.

To add a Rename columns node to your flow diagram

1. Open the menu and then choose Rename columns to add a new transform to your flow diagram, if needed.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema:
 1. Select "Add new rename pair".
 2. Select the current column to rename from the "Current name" column.
 3. Provide a name for the column in the "New name" box.
4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.

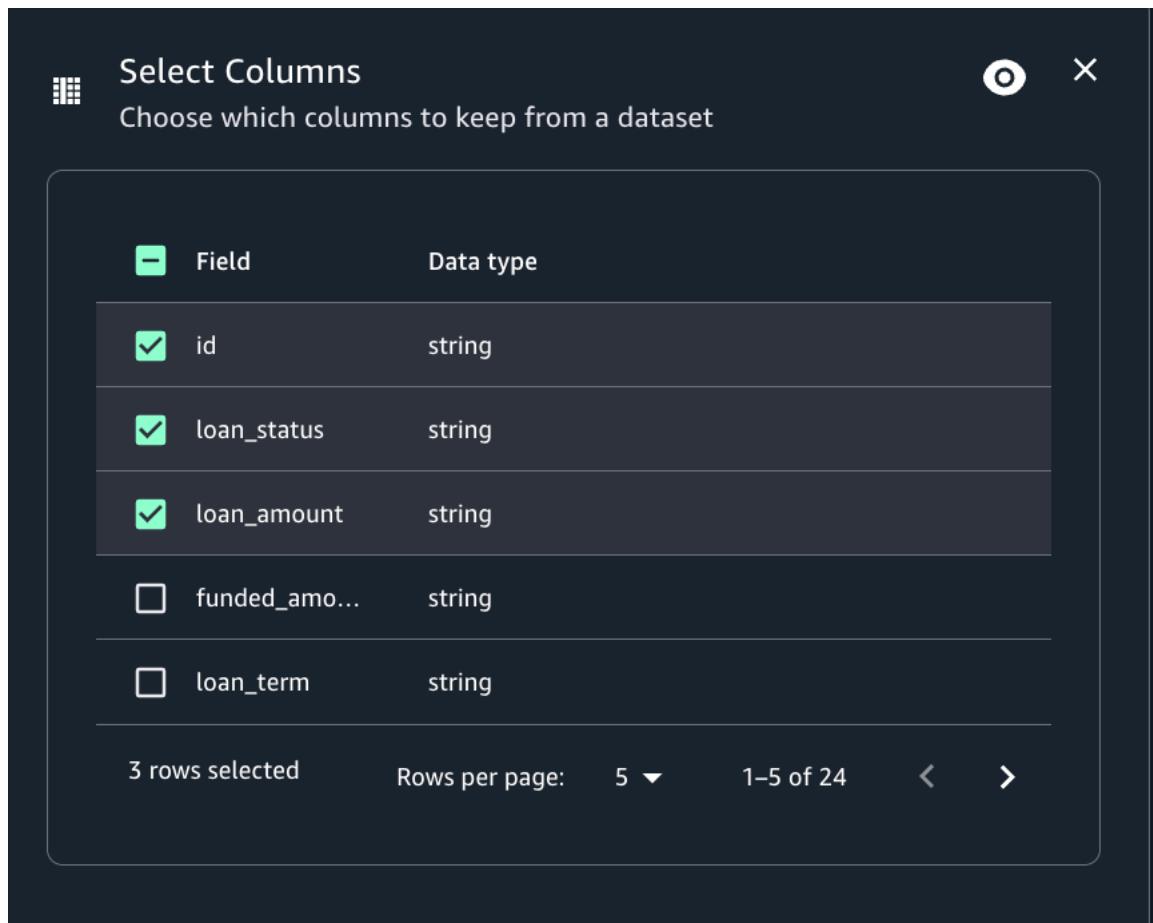


Select columns transform

You can create a subset of columns in a dataset using the Select columns transform. You indicate which columns you want to keep and the rest are removed from the dataset.

To add a **Select columns** node to your flow diagram

1. Open the menu and then choose Filter to add a new transform to your flow diagram, if needed.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema:
 1. Select the columns you want to keep in the output frame by checking the corresponding box under the "Field" option.
 4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.



Split string transform

Use this transform to split a string column into an array.

To add a Split String transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Split String**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **Column to split**, select the column that you want to split.
8. Under **Splitting regular expression**, enter a regular expression that defines the separator token.

9. (Optional) Under **Array column name**, enter a name for the new column that contains the extracted array.

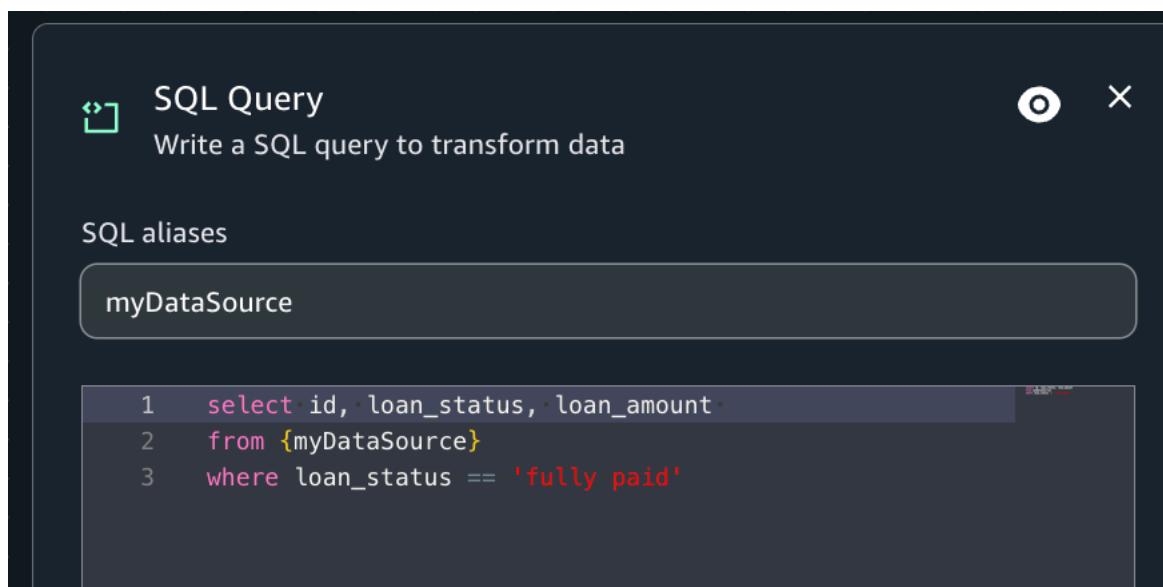
SQL query transform

You can use a SQL query transform to write your own transform in the form of a SQL query.

Writing the SQL query in Visual ETL creates a subset of the data corresponding to the query. A SQL transform node can have multiple datasets as inputs, but produces only a single dataset as output. It contains a text field, where you enter the SQL query.

To add an SQL query node to your flow diagram

1. Open the menu and then choose SQL query to add a new transform to your flow diagram, if needed.
2. (Optional) Click on the rename node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema:
 1. Ensure the alias in the "SQL aliases" box is appropriate. Visual ETL will autopopulate this field, but you can change it.
 2. Write an SQL statement that queries the data to suit your needs
4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.



To timestamp transform

You can use the transform To Timestamp to change the data type of a numeric or string column into timestamp, so that it can be stored with that data type or applied to other transforms that require a timestamp.

To add a To Timestamp transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **To Timestamp**.
4. Select the diagram to add the node to your visual ETL flow.
5. Select the node on the diagram to view details about the transform.
6. Under **Column to convert**, select the column that you want to convert.
7. Under **Time format**, define how to parse the column selected by choosing the format.

If the value is a number, it can be expressed in seconds (Unix/Python timestamp), milliseconds, or microseconds.

If the value is a formatted string, choose the "iso" type. The string needs to conform to one of the variants of the ISO format, for example: "2022-11-02T14:40:59.915Z".

If you don't know the type at this point or different rows use different types, then you can choose "autodetect" and the system will make its best guess, with a small performance cost.

8. Under **Column name**, enter a name for a new column that will be generated for the timestamps. If a column name is not specified, the existing column will be replaced instead of generating a new column.

Union transform

You use the Union transform node when you want to combine rows from more than one data source that have the same schema. When applying Union transformations you can select to "Union by name" to have the union done on columns with the same name (rather than by position). Selecting this option also lets you select "Allow missing columns", which will produce a frame that has missing columns filled with Null values.

To add a Union transform node to your flow diagram

1. Open the menu and then choose "Union transform" to add a new transform to your flow diagram, if needed.
2. (Optional) Click on the union node icon to enter a new name for the node in the flow diagram.
3. Modify the input schema:
 1. Select "Union by name" if you want the union to be done on columns with the same name.
 2. If you have enabled Union by name, select "Allow missing columns" if you want to fill missing columns with null values.
4. (Optional) After configuring the node properties and transform properties, you can preview the modified dataset by choosing the Data preview tab in the node details panel.



Unpivot columns into rows transform

Use this transform to convert column names into values of a new column and generate one row for each column in the original row.

To add an Unpivot Columns Into Rows transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **Unpivot Columns Into Rows**.
4. Select the diagram to add the node to your visual ETL flow.

5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **Unpivot names column**, enter a column to create out of the source column names.
8. Under **Unpivot values column**, enter a column to create out of the source column values.
9. Under **Columns to unpivot into the new value column**, select the column or columns whose names will become the values of the new column.

UUID transform

Use this transform to generate a 36-character unique identifier for each row in your data source.

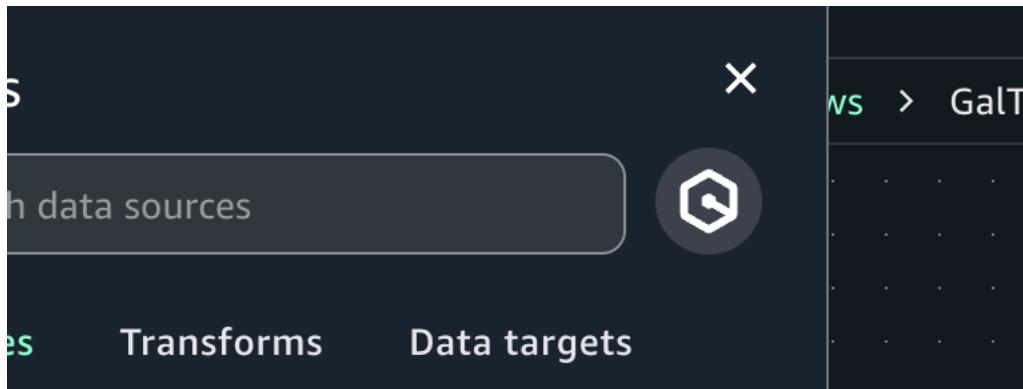
To add a UUID transform:

1. Navigate to your visual ETL flow in Amazon SageMaker Unified Studio.
2. Choose the plus icon to open the **Add nodes** menu.
3. Under **Transforms**, choose **UUID**.
4. Select the diagram to add the node to your visual ETL flow.
5. Connect the transform node to a data source node.
6. Select the node on the diagram to view details about the transform.
7. Under **UUID column name**, enter a name to use for the new column that contains the unique identifiers.

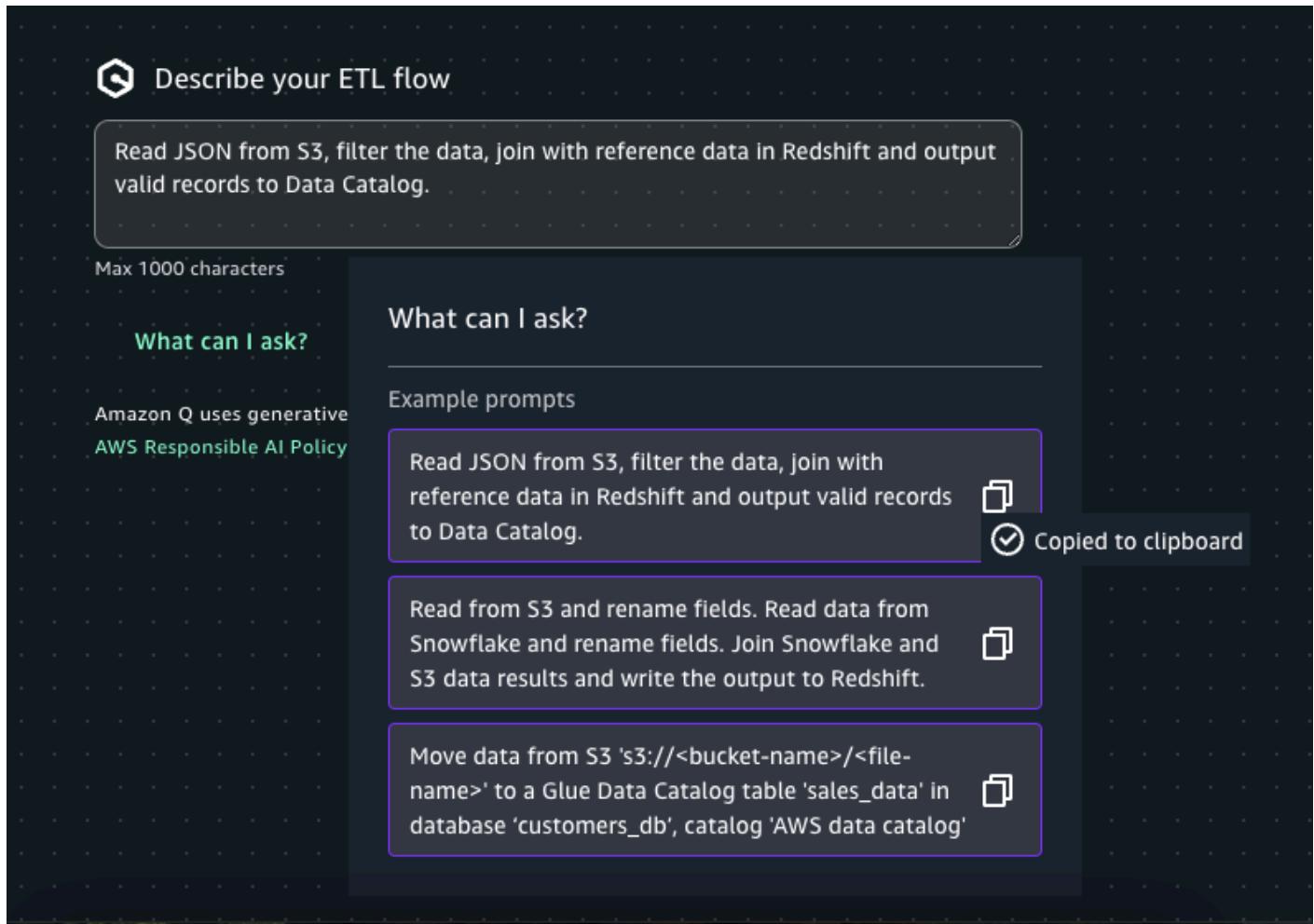
Authoring a visual ETL flow using generative AI

To author a Visual ETL flow using generative AI in Amazon SageMaker Unified Studio:

1. Verify Amazon Q is enabled for your domain.
2. Open the Visual ETL editor.
3. In the "Add nodes" panel click on the Amazon Q icon.



4. (Optional) Click on “What can I ask?” and copy a prompt.
5. Enter the desired prompt in the chat box and click on ‘Submit’.



6. Click on each node in the Visual ETL editor and configure its settings.



Scheduling and running visual flows

There are two ways to schedule Visual ETL flows in Amazon SageMaker Unified Studio.

- You can schedule your visual flows directly in the Visual ETL editor. This way you can schedule a single visual flow quickly.
- You can schedule your visual flow using a DAG and the workflows interface. This way you can combine multiple elements in the same schedule.

Scheduling visual flows from the editor

You can schedule your visual flows to run from within the Visual ETL editor. To do this, use a project with the **All capabilities** project profile or another project profile with scheduling enabled in the Tooling blueprint parameters. If you have created a project that needs to be updated to enable scheduling, contact your admin.

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to your visual ETL flows by choosing **Visual ETL flows** from the **Build** menu.
3. Choose the visual flow you want to schedule from the list to open it in the editor.
4. Choose the Schedule icon in the upper-right corner of the editor.
5. Under **Schedule name**, enter a name for the schedule.
6. Under **Schedule status**, choose an option to determine whether the schedule will begin running after being created.
 - Choose **Active** to activate the schedule and run the Visual ETL flow when the schedule indicates it should run.
 - Choose **Paused** to create a schedule that will not run the Visual ETL flow yet.
7. (Optional) Write a description of the schedule.

8. Choose a schedule type.
 - Choose **One-time** to run the Visual ETL flow at one specific time.
 - Choose **Recurring** to create a schedule that runs the Visual ETL flow at multiple times that you choose.
9. Choose the days and times that the schedule will run.
10. Choose **Create schedule**.

You can then view the schedule on the **Schedules** tab of the Visual ETL page in your project.

Reviewing scheduled visual flows in the editor

You can review scheduled visual flows in the Visual ETL interface in Amazon SageMaker Unified Studio. On the schedules page, you can pause, edit, and delete schedules. You can also view the status and other information for a schedule and choose the name of a schedule to view runs and additional data.

To review scheduled queries, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to your project.
3. Choose **Visual ETL flows** from the **Build** menu.
4. Choose the **Schedules** tab.

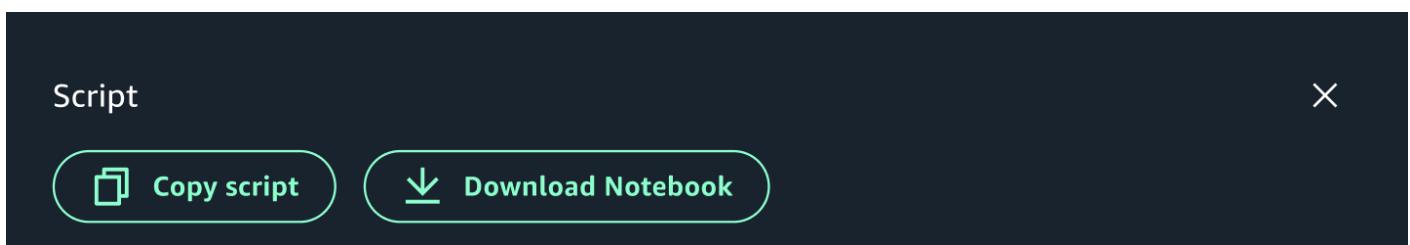
You can then pause, edit, or delete a schedule by choosing the three-dot **Actions** menu next to a schedule in the list.

To view information about different times the schedule has run, choose the name of the schedule to view the **Runs** section for that schedule. You can choose the name of a run to see a log and other details for that run.

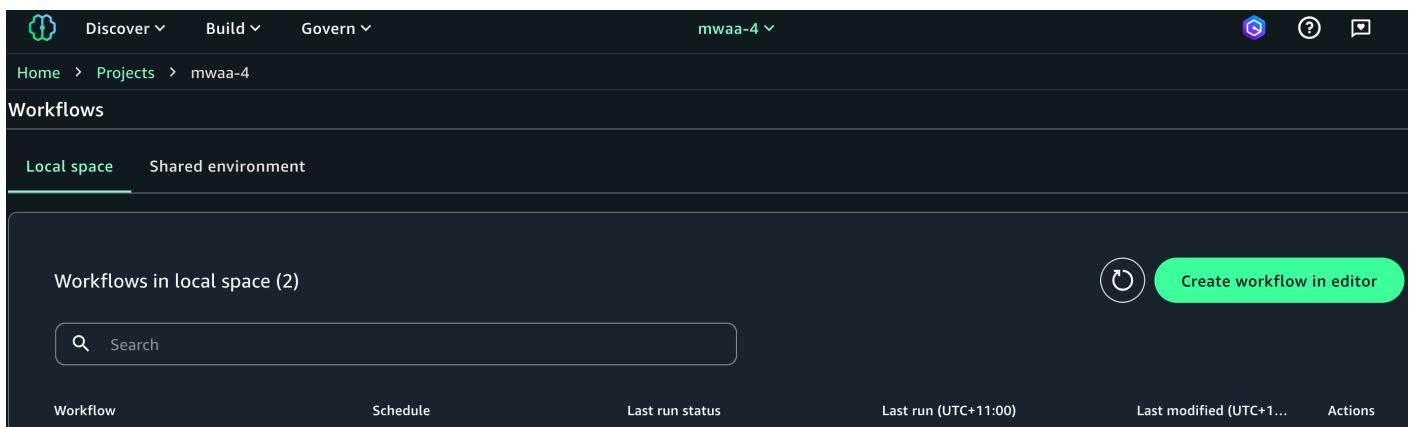
Scheduling visual flows with workflows

You can schedule the Visual ETL flows you authored to run based on a schedule using Workflows. The following is an example of how to do this:

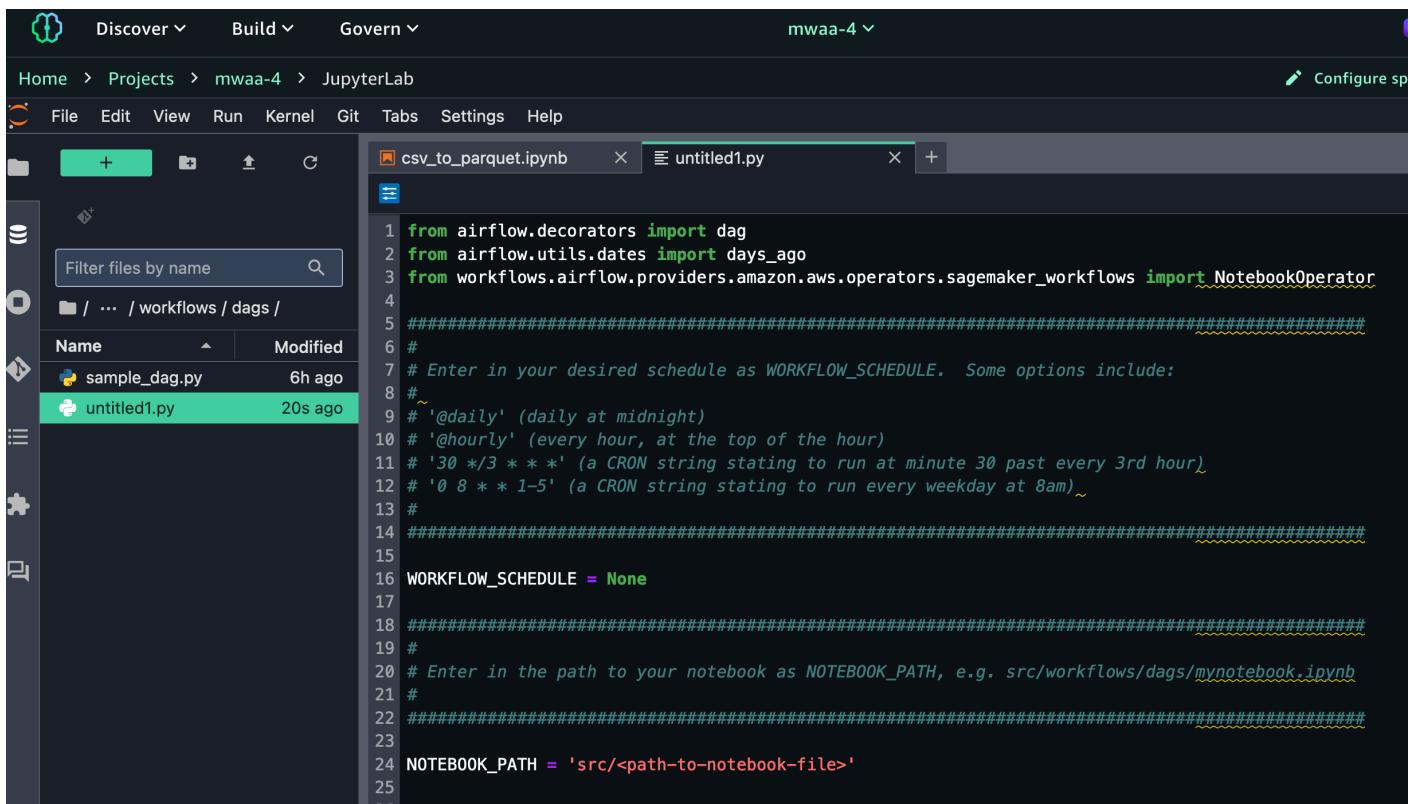
1. Create a Visual ETL flow and name it "mwaa-test".
2. Save your draft flow ("mwaa-test.vetl") to your project.



3. Navigate to Build → Workflows menu, click on the "Create workflow in editor".

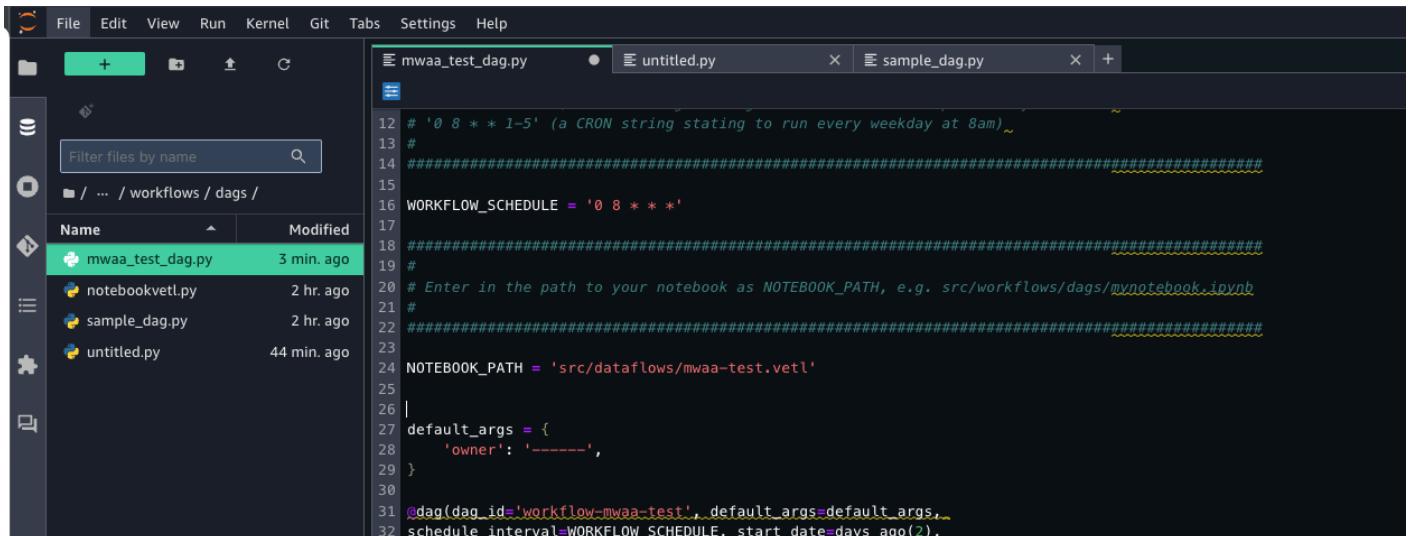


4. You will now see an example DAG template in JupyterLab.



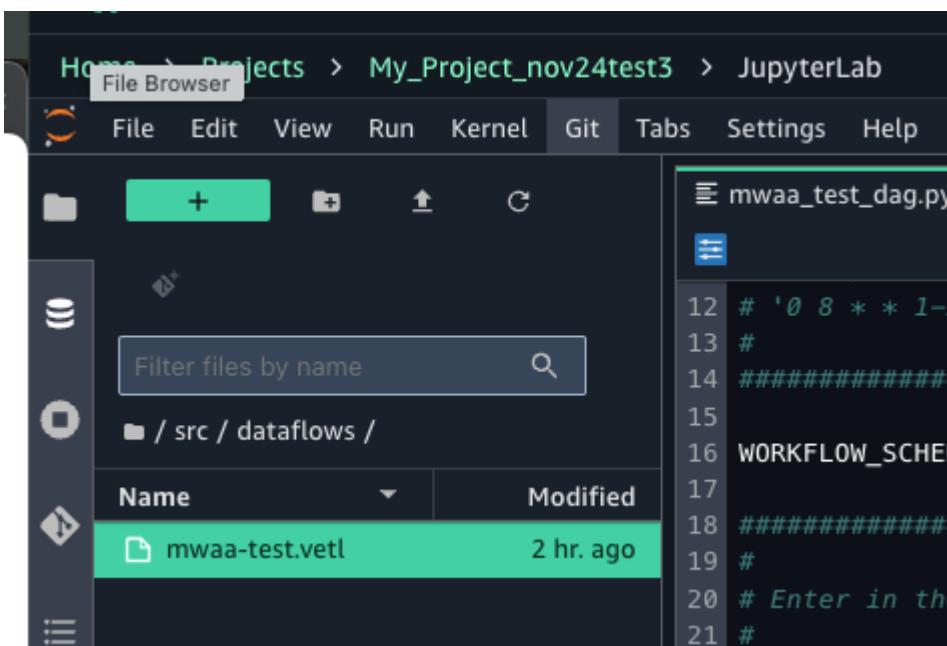
5. Modify the lines of python code as below, then save it as "mwaa_test_dag.py". We will execute the dataflow at 8AM everyday. By default, the dataflow's notebook file is under the path "src/dataflows".

```
WORKFLOW_SCHEDULE = '0 8 * * *'
NOTEBOOK_PATH = 'src/dataflows/mwaa-test.vetl'
dag_id = "workflow-mwaa-test" # optional, set to give your workflow a meaningful name
```



```
12 # '0 8 * * 1-5' (a CRON string stating to run every weekday at 8am)
13 #
14 #####
15
16 WORKFLOW_SCHEDULE = '0 8 * * *'
17
18 #####
19 #
20 # Enter in the path to your notebook as NOTEBOOK_PATH, e.g. src/workflows/dags/mynotebook.ipynb
21 #
22 #####
23
24 NOTEBOOK_PATH = 'src/dataflows/mwaa-test.vetl'
25
26 |
27 default_args = {
28     'owner': '-----',
29 }
30
31 @dag(dag_id='workflow-mwaa-test', default_args=default_args,
32 schedule_interval=WORKFLOW_SCHEDULE, start_date=days_ago(2),
```

6. Pull the file "dataflows/mwaa-test.vetl" from the project's source code repository to JupyterLab.



```
12 # '0 8 * * 1-5'
13 #
14 #####
15
16 WORKFLOW_SCHEDULE = '0 8 * * *'
17
18 #####
19 #
20 # Enter in the path to your notebook as NOTEBOOK_PATH, e.g. src/workflows/dags/mynotebook.ipynb
21 #
22 #####
```



7. Navigate back to the Workflows console, now we can see the DAG is created. We can access Airflow UI via the “Actions” dropdown list.

The screenshot shows the AWS Workflows console under the "mwaa-4" project. The "Workflows" page lists two DAGs: "sample_dag" (Paused) and "workflow-i2xybw9" (At 08:00 AM, Success). A context menu is open for the "workflow-i2xybw9" DAG, showing options: "Pause scheduled workflow", "Run with default parameters", "Run with custom parameters", "Open Airflow UI", and "Edit code".

Workflow	Schedule	Last run status	Last run (UTC+11:00)	Last modified (UTC+11:00)	Actions
sample_dag	Paused	—	—	Nov 22, 2024, 15:59	⋮
workflow-i2xybw9	At 08:00 AM	Success	Nov 23, 2024, 00:10	Nov 22, 2024, 23:58	⋮

8. Manually trigger the DAG.

The screenshot shows the Airflow UI under the "sample" cluster. The "DAGs" page lists two DAGs: "sample_dag" and "workflow-i2xybw9". The "Actions" column for "workflow-i2xybw9" contains a "Trigger DAG" button.

Using both external data and fine-grained data in Amazon SageMaker Unified Studio visual ETL flows

When you use visual ETL, you must select a permission mode to use with your visual ETL flow.

Permission mode is a configuration available to Spark compute resources such as Glue ETL or EMR Serverless. It configures Spark to access different types of data based on the permissions configured for that data. There are two configuration options for permission mode:

- Compatibility mode. This is a configuration for data managed using full-table access, meaning the compute engine can access all rows and columns in the data. Choosing this option enables your compute to work with data assets from AWS and from external systems.
- Fine-grained mode. This is a configuration for data managed using fine-grained access controls, meaning the compute engine can only access specific rows and columns from the full dataset. Choosing this option enables your Glue ETL to work with data asset subscriptions from Amazon SageMaker catalog.

In cases where you want to use both data configured with fine-grained access and data from external sources that you connect to your project, you can use two visual ETL flows and orchestrate them to run together using workflows. To do this, complete the following steps.

Combining flows with different kinds of data in visual ETL

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project you want to use visual ETL in.
3. Choose **Visual ETL** from the **Build** menu.
4. Choose **Create visual ETL flow**.
5. Choose to configure the visual ETL flow with full-table access using the AWS Glue ETL compute named **project.spark.fineGrained**.
6. Configure your visual ETL flow to ingest the subscribed data to an Amazon S3 target used for temporary staging. This can be done by using the plus icon and adding an Amazon SageMaker Lakehouse node as a data source and an Amazon S3 node as a data target, then connecting the nodes on the diagram.
7. Select the Amazon SageMaker Lakehouse node and configure it to point to the data you want to use.

- a. Under **Database**, choose the name of the database you want to use.
 - b. Under **Table**, choose the name of the table you want to use.
8. Configure the Amazon S3 node to point to a new location.
- a. Under **S3 URI**, create a new Amazon S3 folder name and note the location for later use.
 - b. Under **Mode**, select **Overwrite** to clear the Amazon S3 bucket and overwrite it with new data when you are ready to use it again.
 - c. (Optional) Configure the other settings as desired.
9. Save the flow and run it using **project.spark.fineGrained** to verify correctness of the results.
10. Create a new visual ETL flow that uses the AWS Glue ETL compute named **project.spark.compatibility**.
11. Configure this second visual ETL flow to combine the data from the staging S3 location and the data accessible through full-table access to generate the final result.
- a. Select the plus icon. Under **Data sources**, select Amazon S3 and place the node on the diagram.
 - b. Select the Amazon S3 node to configure it.
 - c. Under **S3 URI**, enter the Amazon S3 folder location you used in the first visual ETL flow.
 - d. Use the plus icon, and under **Data sources**, select an external data source to add to your visual ETL flow. Place the node on the diagram.
 - e. Use the plus icon to add a data target and place the data target node on the diagram.
 - f. Select the external data source and the data target to edit the configurations as desired and point to the locations you want to use.
 - g. Use the plus icon, and under **Transforms**, select the **Join** transform. Place the transform on your diagram.
 - h. Connect the Amazon S3 node containing the data from the first flow and the other data source to the data target using the **Join** transform.
12. Save the second flow and run it using **project.spark.compatibility** to verify correctness of the results.
13. Orchestrate these two visual ETL flows using Amazon SageMaker Unified Studio workflows. For more information, see [the section called “Scheduling and running visual flows”](#) and [the section called “Create a workflow”](#).

Make sure that the workflow is configured so that the first visual ETL flow finishes running before the second visual ETL flow runs. By default, they'll run in succession, one after the other. This can also be configured using the `wait_for_completion` param, as shown in [the section called "Sample workflow"](#).

Best practices for Visual ETL in Amazon SageMaker Unified Studio

To get the most out of Visual ETL in Amazon SageMaker Unified Studio:

- Start with simple flows and gradually increase complexity as you become more familiar with the tool.
- Use data preview features frequently to verify the results of your transformations.
- Leverage custom transformations to standardize and streamline your flows.
- Monitor flows performance and optimize as necessary, using Amazon SageMaker Unified Studio's built-in performance analytics.

By following these guidelines and exploring the various features of Visual ETL, you can efficiently create powerful data integration and transformation Visual ETL flows in Amazon SageMaker Unified Studio.

Using Amazon Q with Amazon SageMaker Unified Studio

Amazon SageMaker Unified Studio has an integration with Amazon Q with which you can interact with AWS resources through a conversational interface. You can ask questions about your assets in Amazon SageMaker Unified Studio using simple, natural language queries within the Amazon Q chat interface.

To start using Amazon Q to chat about your project and your assets:

1. Log in to your AWS account and navigate to Amazon SageMaker Unified Studio.
2. Select the the Amazon Q chat interface within Amazon SageMaker Unified Studio.
3. Begin typing your query in natural language, asking Q to list Amazon SageMaker AI catalog assets published in the catalog. For example, you can say: "Find me data on sales".

Amazon Q interprets your query, executes the appropriate API calls, and returns the results directly in the chat interface.

Machine learning

Amazon SageMaker Unified Studio is a unified development experience for building analytics, AI/ML, and generative AI applications at scale. This chapter describes the Amazon SageMaker AI capabilities that you can use in Amazon SageMaker Unified Studio.

Note

When you add a custom tag to a SageMaker AI resource (such as a training job, inference endpoint, model, or pipeline), add the prefix `ProjectUserTag` to the tag name. For example:

```
ProjectUserTagMyCustomTag
```

Note

ECR repositories must be created with the `AmazonDataZoneProject` tag with the project ID (which can be found under project details in the project overview page or from the page URL) as the tag value. If you want to add your own tags, they must be prefixed with `ProjectUserTag`.

For example, with AWS CLI:

```
aws ecr create-repository \
--repository-name my-repo \
--tags \
Key=AmazonDataZoneProject,Value=5blxelum5cmckb \
Key=ProjectUserTagMyTag,Value=MyTagValue \
```

Example using Jupyterlab notebook:

```
import boto3

# Create ECR client
ecr_client = boto3.client('ecr')
```

```
# Define repository name
repository_name = 'my-ecr-repo'

# Define tags
tags = [
    {
        'Key': 'AmazonDataZoneProject',
        'Value': '5blxelum5cmckb'
    },
    {
        'Key': 'ProjectUserTagMyTag',
        'Value': 'MyTagValue'
    },
]

try:
    # Create the repository with tags
    response = ecr_client.create_repository(
        repositoryName=repository_name,
        imageScanningConfiguration={
            'scanOnPush': True
        },
        encryptionConfiguration={
            'encryptionType': 'AES256'
        },
        tags=tags
    )

    repository_uri = response['repository']['repositoryUri']
    print(f"Repository created successfully!")
    print(f"Repository URI: {repository_uri}")

except ecr_client.exceptions.RepositoryAlreadyExistsException:
    print(f"Repository {repository_name} already exists")
    # Add tags to existing repository
    ecr_client.tag_resource(
        resourceArn=f"arn:aws:ecr:{ecr_client.meta.region_name}:
{boto3.client('sts').get_caller_identity()['Account']}:{repository/
{repository_name}}",
        tags=tags
    )
    # Get the repository URI
```

```
response =  
    ecr_client.describe_repositories(repositoryNames=[repository_name])  
    repository_uri = response['repositories'][0]['repositoryUri']  
    print(f"Added tags to existing repository")  
    print(f"Repository URI: {repository_uri}")  
except Exception as e:  
    print(f"Error creating repository: {str(e)}")
```

ECR repositories without the AmazonDataZoneProject cannot be used. You must create new ECR repositories with the AmazonDataZoneProject tag. Once tagged with the AmazonDataZoneProject tag, this tag cannot be modified or removed from your ECR repositories. For more information about ECR repositories, see <https://docs.aws.amazon.com/AmazonECR/latest/userguide/what-is-ecr.html>.

Topics

- [Discover Jumpstart models](#)
- [Build models in JupyterLab](#)
- [Train models](#)
- [Use inference endpoints to deploy models](#)
- [Pipelines](#)
- [Model registry](#)
- [Track experiments using MLflow](#)
- [HyperPod clusters](#)
- [Partner AI apps](#)

Discover Jumpstart models

Amazon SageMaker Unified Studio maintains publicly available foundation models for you to access, customize, and integrate into your machine learning lifecycles. A foundation model is a large pre-trained model that's adaptable to a variety of downstream tasks and often serves as the starting point for developing more specialized models.

To explore the available models from our model providers, follow these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the main menu, choose **Build**.
3. From the drop-down menu, choose **Jumpstart Models**.

The system opens a page that lists the model providers.
4. Choose a provider.
5. From the provider's list of models, choose a model to view its details.

For more information about discovering models, see [JumpStart foundation models](#) in the *Amazon SageMaker AI Developer Guide*.

Build models in JupyterLab

Use the JupyterLab space within Amazon SageMaker Unified Studio to run JupyterLab applications. A JupyterLab space is a private or shared space that manages the storage and compute resources needed to run the JupyterLab application.

The JupyterLab application is a web-based interactive development environment (IDE) for notebooks, code, and data. Use the JupyterLab application's flexible and extensive interface to configure and arrange machine learning (ML) workflows.

Your project contains a configured JupyterLab space.

To open the JupyterLab space in Amazon SageMaker Unified Studio, follow these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the main menu, choose **Build**.
3. Under **IDE & Applications**, choose **JupyterLab**.

Amazon SageMaker Unified Studio opens the JupyterLab space associated with your project. Choose **Configure space** to tailor the configuration to your needs.

For more information about using JupyterLab, see [SageMaker JupyterLab](#) in the *Amazon SageMaker AI Developer Guide*.

Train models

Using Amazon SageMaker Unified Studio, you can train foundation models or custom models.

Follow these steps to train a foundation model:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. Choose a model to train.

- a. From the main menu, choose **Build**.
- b. From the drop-down menu, choose **Jumpstart Models**.

The JumpStart page lists the model providers.

- c. Choose a model provider. The page displays the models for that provider.
- d. Under **Action**, choose **Trainable**. The page displays the trainable models for that provider.
- e. From the provider's list of models, choose the model you want to train.

3. From the model details page, choose **Train** to create a training job.

If the model is pretrained, you can fine-tune the model by adjusting the model parameters.

4. In the **Fine-tuning model** page, update the hyperparameters you want to change.
5. Enter **Submit** to submit the training job. You can view the training job from the **Training jobs** page.

You can also train the model in a Jupyterlab notebook using the SageMaker AI python SDK.

For more information about training models in JumpStart, see [JumpStart pretrained models](#) in the *Amazon SageMaker AI Developer Guide*.

Use inference endpoints to deploy models

Endpoint are locations where you send inference requests to your deployed machine learning models. After you create an endpoint, you can add models to it, test it, and change its settings as needed. By using endpoints, you don't have to manage the underlying infrastructure for configuring and deploying a model.

For more information about using endpoints for real-time inference, see [Deploy models for real-time inference](#) in the *Amazon SageMaker AI Developer Guide*. Also see the [Getting started with deploying real time models on SageMaker AI](#) blog post.

Topics

- [Create an endpoint and deploy a model](#)
- [View your endpoints](#)

Create an endpoint and deploy a model

To create an endpoint, follow these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the main menu, choose **Build**.
3. From the drop-down menu, choose **Inference endpoints**.
4. From the **Endpoints** page, choose **Create endpoint**.
5. From the **Create endpoint** page, configure these values:
 - For **Endpoint name**, enter a name for the endpoint.
 - For **Instance type**, choose an instance for the endpoint.
 - For **Initial instance count**, enter the number of instances for the endpoint to provision initially.
 - For **Maximum instance count**, enter the maximum number of instances that the endpoint can provision, when it scales up.
6. Under **Models**, choose **Add model**. In the **Add model** modal form, follow these steps:
 - a. Select the model type (JumpStart foundation models or Deployable models that you created).

The form lists the models that are compatible with the instance type you selected.

 - b. Choose one of the models.
 - c. Under **Model settings**, enter these values:
 - Number of CPU cores – Number of accelerators to deploy.
 - Minimum number of copies – minimum number of model copies to deploy.

- Min CPU memory – Minimum amount of CPU memory.
 - Max CPU memory – Maximum amount of CPU memory.
- d. Choose **Add model**.
7. Choose **Deploy** to deploy the endpoint.

View your endpoints

To view your endpoints in the **Endpoints** table, follow these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the main menu, choose **Build**.
3. From the drop-down menu, choose **Inference endpoints**.
4. (Optional) To search for specific endpoints, enter text in **Search by endpoint name**.

Pipelines

Amazon SageMaker Unified Studio supports SageMaker AI Pipelines, a workflow orchestration service for automating machine learning (ML) development.

A pipeline defines a series of interconnected steps in a directed acyclic graph (DAG). You can define the steps using the Amazon SageMaker Unified Studio visual pipeline designer, or by creating a pipeline definition JSON schema. This DAG JSON definition gives information on the requirements and relationships between each step of your pipeline.

The structure of a pipeline's DAG is determined by the data dependencies between steps. These data dependencies are created when the properties of a step's output are passed as the input to another step.

Note

To add a custom tag to a pipeline, add the prefix `ProjectUserTag` to the tag name. For example:

`ProjectUserTagMyCustomTag`

For an overview of pipelines, see [Pipelines overview](#) in the *Amazon SageMaker AI Developer Guide*.

Pipeline actions

The following sections describe the actions available in Amazon SageMaker Unified Studio to create and manage pipelines:

Topics

- [Define a pipeline](#)
- [Edit a pipeline](#)
- [Run a pipeline](#)
- [Stop a pipeline execution](#)
- [View the details of a pipeline](#)
- [View the details of a pipeline run](#)
- [Download a pipeline definition file](#)

Define a pipeline

You define a pipeline using the visual pipeline designer. You can also create a [pipeline definition JSON schema](#).

To define a pipeline using the visual pipeline designer, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose **Create in visual editor**. The system opens the visual editor for creating a pipeline. You can also import a pipeline definition file from your computer.
4. Use the visual editor to add and connect pipeline steps.
5. Choose the **Save** to save your changes.

For more details, see [Define a pipeline](#) in the *Amazon SageMaker AI Developer Guide*.

Edit a pipeline

You can make changes to a pipeline before running it. To edit a pipeline, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose the pipeline to edit.
4. Choose the **Executions** tab.
5. Choose the pipeline execution to edit.
6. Choose the **Visual editor** to edit the pipeline.
7. Choose the **Save** to save your changes.

For more details, see [Edit a pipeline](#) in the *Amazon SageMaker AI Developer Guide*.

Run a pipeline

After defining the steps of your pipeline as a directed acyclic graph (DAG), you can run your pipeline, which executes the steps defined in your DAG.

To run a pipeline, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose the pipeline to run.
4. Choose **Execute**.
 - a. For **Execution name**, enter a name for this run.
 - b. (Optional) For **Description**, enter a description for this run.
5. Choose **Execute** to start the run.

For more details, see [Run a pipeline](#) in the *Amazon SageMaker AI Developer Guide*.

Stop a pipeline execution

To stop a pipeline, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose the pipeline to stop.
4. Choose the **Executions** tab.
5. Choose the execution to stop.
6. Choose **Stop** to stop the execution. To resume the execution from where it was stopped, choose **Resume**.

View the details of a pipeline

You can view the details of a pipeline to understand its parameters, the dependencies of its steps, or monitor its progress and status.

To access the details of a given pipeline using Amazon SageMaker Unified Studio, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose the pipeline to view its details.
4. Choose any of the following tabs to view these details:
 - **Executions** – Details about the executions.
 - **Graph** – The pipeline graph, including all steps.
 - **Parameters** – The run parameters and metrics related to the pipeline.
 - **Information** – The metadata associated with the pipeline, such as tags, the pipeline Amazon Resource Name (ARN), and role ARN. You can also edit the pipeline description from this location.

View the details of a pipeline run

You can view the details of a pipeline run, which can help you:

- Identify and resolve problems that may have occurred during the run, such as failed steps or unexpected errors.
- Compare the results of different pipeline executions to understand how changes in input data or parameters impact the overall workflow.
- Identify bottlenecks and opportunities for optimization.

To view the details of a pipeline run, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose the pipeline to view its details.
4. Choose the **Executions** tab.
5. Choose the pipeline execution to view. The pipeline graph for that execution appears.
6. Choose any of the pipeline steps in the graph to see step settings in the right sidebar.
7. Choose any of the following tabs to view these details:
 - **Graph** – The pipeline graph, including all steps.
 - **Parameters** – The run parameters and metrics related to the pipeline.
 - **Information** – The metadata associated with the pipeline, such as tags, the pipeline Amazon Resource Name (ARN), and role ARN. You can also edit the pipeline description from this location.

Download a pipeline definition file

You can download the definition file for your pipeline. You can use this pipeline definition file for:

- **Backup and restoration:** Use the downloaded file to create a backup of your pipeline configuration, which you can restore in case of infrastructure failures or accidental changes.
- **Version control:** Store the pipeline definition file in a source control system to track changes to the pipeline and revert to previous versions if needed.

- **Programmatic interactions:** Use the pipeline definition file as input to the SDK or AWS CLI.
- **Integration with automation processes:** Integrate the pipeline definition into your CI/CD workflows or other automation processes.

To download the definition file of a pipeline, complete the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **ML Pipelines**. The system displays the pipelines for your project.
3. Choose the pipeline. You can download the pipeline definition from this page or any of the execution pages.
4. At the top right of the page, choose the vertical ellipsis and choose **Download pipeline definition (JSON)**.

For more information about the pipeline actions, see [Pipelines actions](#) in the *Amazon SageMaker AI Developer Guide*.

Model registry

Use the model registry to catalog your models and manage model deployment to production.

You catalog models by creating model (package) groups that contain different versions of a model. You can create a model group that tracks all the models that you train to solve a particular problem. You can then register each model you train and the model registry adds it to the model group as a new model version.

You can create categories of model groups by organizing them into collections. A typical workflow might include the following tasks:

1. Create a model group.
2. Create an ML pipeline that trains a model. For information about pipelines, see [Pipelines](#).
3. For each run of the ML pipeline, create a model version that you register in the model group you created in the first step.
4. Add your model group to one or more collections.

For details about how to work with the model registry, see [Model Registry, Model Versions, and Model Groups](#) in the *Amazon SageMaker AI Developer Guide*.

Create a model group

A model group contains different versions of a model. Follow these steps to create a model group:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **Model Registry**. The model registry page displays the models that are registered to your project.
3. Choose **Model Groups**. The page displays the model groups that are defined for your project.
4. From the actions menu, choose **Create model group**.
5. Provide a name for the model group. Optionally, you can add keys to the model group.
6. Choose **Register model group** to create the model group.
7. Confirm that your newly-created model appears in the list of model groups.

Create a collection

Follow these steps to create a collection:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **Model Registry**. The model registry page displays the models that are registered to your project.
3. Choose **Collections**. The collections page displays the collections that are defined for your project.
4. In the **Actions** drop-down menu, choose **Create collection**.
5. Provide a name for the collection.
6. (Optional) To add model groups to the collection, complete these steps:
 - a. Choose **Select model groups**.
 - b. Select up to 10 model groups that you want to add.
7. Choose **Create** to create the collection.

Register a model version

The model registry is structured as several model (package) groups with model packages in each group. Each model package in a model group corresponds to a trained model. The version of each model package is a numerical value that starts at 1 and is incremented with each new model package added to a model group. The model packages used in the model registry are versioned, and **must** be associated with a model group.

Follow these steps to register a model version:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **Model Registry**. The model registry page displays the models that are registered to your project.
3. Choose **Register**.
4. From the **Register Model** page, Choose the type of model artifact:
 - JumpStart – Choose from the list of models.
 - Jobs – Choose a training job
 - Bring-your-model – Enter the location of your models
5. Choose **Register**.
6. Create a model group, or find an existing model group.
7. Choose **Register**.

For more information, see [Model Registry](#) in the *Amazon SageMaker AI Developer Guide*.

Track experiments using MLflow

Use MLflow in Amazon SageMaker Unified Studio to create, manage, analyze, and compare machine learning experiments.

 **Note**

Amazon SageMaker AI MLflow dataplane API operations don't support AWS CloudTrail logs.

For more information about MLflow, see [Machine learning experiments using MLflow](#) in the *Amazon SageMaker AI Developer Guide*.

Topics

- [MLflow Tracking Servers](#)
- [Tracking experiments with MLflow](#)

MLflow Tracking Servers

MLflow uses compute and storage resources provided by an MLflow Tracking Server. Each project requires an MLflow Tracking Server. Your domain administrator can configure the project defaults to automatically create the MLflow Tracking Server during project creation. Otherwise, you can create an MLflow Tracking Server on demand for the project.

When you delete a project, Amazon SageMaker Unified Studio automatically deletes the tracking server.

For more information about MLflow Tracking Servers, see [MLflow Tracking Servers](#) in the *Amazon SageMaker AI Developer Guide*.

For more information about project profiles for AI-ML projects, see [Project profiles](#) in the *Amazon SageMaker Unified Studio Admin Guide*.

Create the MLflow tracking server

After you create a project, you can create the MLflow Tracking Server for the project, if it wasn't created automatically during project creation.

To create an MLflow Tracking Server, perform the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the top banner, choose your project from the projects drop-down menu, and choose **Project overview**.
3. From the left menu, choose **Compute**.
4. From the tabs in the top banner, choose **MLflow Tracking Servers**.
5. Choose **Create MLflow Tracking Server**.
6. (Optional) Provide values to override the default values for the following fields:
 - a. Name – enter a name for the server.
 - b. Size – select a size for the server.
7. Choose **Create** to create the server.

Edit the MLflow Tracking Server

After you create a tracking server, you can change the configured server size, if the current size isn't sufficient for the project.

To edit a tracking server, perform the following steps, starting at your project's **MLflow Tracking Servers** page:

1. From the **Actions** drop-down menu, choose **Edit**. You can change the following values:
 - a. Size – select a new size for the server.
 - b. Artifact storage S3 path – enter a new path to the artifact storage.
2. Choose **Save changes** to update the tracking server.

Start or stop an MLflow server

You can stop a running server or start a stopped server. While the tracking server is starting or stopping, it's not available for MLflow to use.

To start or stop an MLflow tracking server, perform the following steps from your project's **Project details** page:

1. From the left menu, choose **Compute**.
2. From the tabs in the top banner, choose **MLflow Tracking Servers**.
3. From the **Actions** drop-down menu, choose **Stop** to stop a running server. Choose **Start** to start a stopped server.

Integrate MLflow with your environment

For information about how to integrate MLflow with your environment, see [Integrate MLflow with your environment](#) in the *Amazon SageMaker AI Developer Guide*.

Launching MLflow UI

You can launch MLflow Tracking Server UI from the **MLflow Tracking Servers** page, by performing the following steps:

1. Navigate to the project details page for your project.
2. From the left menu, choose **Compute**.

3. From the tabs in the top banner, choose **MLflow Tracking Servers**.
4. From the **Actions** drop-down menu, choose **Open MLflow**. This action uses a presigned URL to launch MLflow UI in a new tab in your current browser.

For more information, see [Launch the MLflow UI using a presigned URL](#) in the *Amazon SageMaker AI Developer Guide*.

Tracking experiments with MLflow

From the **Experiments** page, you can view and track the experiments for the current project. You can also open sample notebooks, such as for logging MLflow experiments and for registering MLflow models.

View the list of experiments

To view the list of experiments in your project, perform the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **MLflow**. The **Experiments** page displays the MLflow experiments for your project.
3. (Optional) Enter text in the search text box to view a subset of the listed experiments.

Track an experiment

To track an MLflow experiment in your project, perform the following steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **MLflow**. The **Experiments** page displays the MLflow experiments for your project.
3. From the **Experiments** table, choose the experiment to track. This launches a new MLflow tab in your current browser.

HyperPod clusters

Use Amazon SageMaker AI HyperPod to help you provision resilient compute clusters for running model training or fine-tuning workloads. Amazon SageMaker AI HyperPod integrates with Slurm or Amazon EKS for orchestration.

You can create HyperPod clusters using the Amazon SageMaker AI Hyperpod console UI or SageMaker AI Studio. For more information, see [Orchestrating SageMaker AI HyperPod clusters with Slurm](#) or [Orchestrating SageMaker AI HyperPod clusters with Amazon EKS](#) in the *Amazon SageMaker AI Developer Guide*.

In Amazon SageMaker Unified Studio, you can launch machine learning workloads on Amazon SageMaker AI HyperPod clusters. You can also view details about the HyperPod clusters.

Topics

- [Connect to a HyperPod cluster](#)
- [View the HyperPod clusters](#)
- [View details about a HyperPod cluster](#)
- [HyperPod task governance](#)
- [Open the HyperPod in JupyterLab](#)

Connect to a HyperPod cluster

To use a HyperPod cluster in Amazon SageMaker Unified Studio, you create a connection to the cluster by following these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **HyperPod**. The compute page displays the HyperPod clusters for your project.
3. Choose **Add compute**.
4. In the **Add compute** form, configure the following fields:
 - a. For **Connection name**, enter a name for this connection.
 - b. For **HyperPod cluster name**, enter the name of the HyperPod cluster.
 - c. For **Access role ARN**, enter the IAM role that the project needs to assume.
 - d. For **Account ID**, enter the AWS account where the runtime role exists.
 - e. For **AWS Region**, enter the Region where the HyperPod cluster was created.

View the HyperPod clusters

To view the HyperPod clusters in your project, follow these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Build** drop-down menu, choose **HyperPods**.

The portal opens the **HyperPod clusters** tab of the **Compute** page. The HyperPod clusters table provides a summary view of each cluster, including the ARN, status, and creation time.

View details about a HyperPod cluster

To view the details page for a HyperPod cluster, choose the HyperPod from the table of HyperPod clusters. The page displays tabs for tasks, metrics, settings, and metadata details.

For more information about HyperPod cluster details that you can view in Amazon SageMaker Unified Studio, see [HyperPod tabs in Studio](#) in the *Amazon SageMaker AI Developer Guide*.

HyperPod task governance

For Amazon EKS clusters, you can use HyperPod task governance to streamline resource allocation and utilization of compute resources in the cluster.

HyperPod task governance provides a comprehensive dashboard view of your Amazon EKS cluster utilization metrics, including hardware, team, and task metrics.

For more information about the HyperPod dashboard view, see [Dashboard](#) in the *Amazon SageMaker AI Developer Guide*.

Open the HyperPod in JupyterLab

To open your HyperPod in JupyterLab, follow these steps:

1. From the cluster details page, choose **Open in JupyterLab**.

The **Starting space** page opens and the space initialization starts.

After the JupyterLab space is ready, it opens the HyperPod sample notebook.

2. The HyperPod sample notebook shows the end-to-end flow of how to use the HyperPod cluster, including sample commands for:
 - Connecting to the cluster
 - Submitting jobs to the cluster.

- Viewing job status or cluster status.

Partner AI apps

Amazon SageMaker Unified Studio provides access to Amazon SageMaker AI Partner AI apps. Partner AI apps include generative AI and machine learning (ML) development applications that are built, published, and distributed by industry-leading application providers.

Amazon SageMaker AI Partner AI apps are full application stacks that include an Amazon EKS cluster and accompanying services such as Application Load Balancer, Amazon RDS, Amazon S3 buckets, or Amazon SQS queues.

To view the partner AI apps, complete these steps:

1. Sign in to Amazon SageMaker Unified Studio using the link that your administrator gave you.
2. From the **Explore** drop-down menu, choose **Partner AI apps**.

For more information about SageMaker AI partner AI apps, see [Partner AI Apps overview](#) in the *Amazon SageMaker AI Developer Guide*.

Amazon Bedrock in SageMaker Unified Studio

With Amazon Bedrock in SageMaker Unified Studio you can build generative AI apps that use Amazon Bedrock models and features, such as knowledge bases and guardrails, without needing to write any code.

To use Amazon Bedrock in SageMaker Unified Studio, you must be a member of an Amazon SageMaker Unified Studio domain. Your organization will provide you with login details. Contact your administrator if you don't have your login details.

Your organization's administrator determines which Amazon Bedrock models and features you have access to. Contact your organization's administrator if you need access to a model or feature that you don't currently have access to.

Note

If you are administrator and need information about managing Amazon Bedrock in SageMaker Unified Studio, see [Amazon Bedrock in SageMaker Unified Studio](#) in the *Amazon SageMaker Unified Studio admin guide*.

Discover Amazon Bedrock in SageMaker Unified Studio playgrounds

Amazon Bedrock in SageMaker Unified Studio provides various options for discovering and experimenting with Amazon Bedrock models and apps.

With the model catalog you can find information about the Amazon Bedrock models that are available to you and decide which model is suitable for your use case. Different models have different capabilities and modalities. For more information, see [Find serverless models with the Amazon Bedrock model catalog](#).

Amazon Bedrock in SageMaker Unified Studio offers two playgrounds for you to experiment with Amazon Bedrock models in: the [chat](#) playground and the [image and video](#) playground. With the chat playground, you can generate text responses from a model by sending text and image prompts. You can also interact with chat agent apps that have been shared with you. With the image and video playground, you can generate and edit images and videos by sending text and

image prompts to a suitable model. For more information, see [Experiment with the Amazon Bedrock playgrounds](#).

Build generative AI apps

Within an Amazon SageMaker Unified Studio project, you can create two types of generative AI apps: a [chat agent app](#) and a [flow app](#). You can use a chat agent app to chat with an Amazon Bedrock model through a conversational interface, typically by sending prompts (text or image) and receiving responses. You can use a flows app to link prompts, supported Amazon Bedrock models, and other units of work, such as a knowledge base, together and create generative AI workflows.

Apps that you create with Amazon Bedrock in SageMaker Unified Studio can integrate the following Amazon Bedrock features.

- [**Data sources**](#) — Enrich apps by including context that is received from querying a knowledge base or a document.
- [**Guardrails**](#) — Implement safeguards for your Amazon Bedrock in SageMaker Unified Studio app based on your use cases and responsible AI policies.
- [**Functions**](#) — Call a function with a model to access a specific capability when handling a prompt.
- [**Prompts**](#) — Access reusable prompts that you can use in a flow app.

Within a project, you can use the *asset gallery* to organize the prompts and components that you use for an app. A component is an Amazon Bedrock knowledge base, guardrail, or function.

A critical part of creating a generative AI app is deciding which model to use and which model settings to use. To help you decide, you can [evaluate](#) a model for different task types.

If you work on a team, you can collaborate by [sharing](#) an app with other team members. You can also [export](#) an app so that you can use the app in your own environment.

You can clone the repository that holds your Amazon SageMaker Unified Studio project files to your computer. However, we don't recommend making changes to your project's files on your local desktop, as this may break your project's apps and components.

Find serverless models with the Amazon Bedrock model catalog

The Amazon Bedrock in SageMaker Unified Studio model catalog is where you can find the serverless Amazon Bedrock foundation models that you have access to. You can group models by their modality or by their provider. The modality of a model represents the type of input data that the model is trained on and is able to process, such as text or image data. For more information, see [Supported foundation models in Amazon Bedrock](#).

If you can't find a specific model, ask your administrator if you have permissions to access the model.

To find out information about a model, such as supported use cases, select the model tile in the model catalog. When choosing a model, consider the following:

- Amazon Bedrock models support differing inference parameters and capabilities. For more information, see [Amazon Bedrock foundation model information](#) in the *Amazon Bedrock user guide*.
- Amazon Bedrock in SageMaker Unified Studio supports Amazon Bedrock foundation models with on-demand throughput and [cross-region inference](#).

Models that support cross-region inference throughput can increase throughput and improve resiliency by sending requests to different AWS Regions during peak utilization bursts. In the model catalog (and the model selector in app configuration), the text *Cross-region* identifies such a model.

- Amazon Bedrock in SageMaker Unified Studio doesn't support [Provisioned throughput, custom models](#) or [imported models](#).

If the model is suitable for your needs, you can choose the menu button to start using the model. Depending on the model, you can choose from the following actions:

- [Build chat agent app](#) – Create an app in which users can chat with a model.
- [Build flow app](#) – Visually create the workflow for an app.
- [Build prompt](#) – Create reusable prompts for use in a flow app.
- [Evaluate model](#) – Evaluate the performance of a model for your use case.

The following procedure shows how to open the model catalog from the Amazon Bedrock in SageMaker Unified Studio playground. You can also access the model catalog from your projects.

Your administrator might give you access to different models in your projects. To check the models that you can access in a project, open or create a project, and then select **Models** in the navigation pane to open the model catalog.

To open the model catalog in the playground

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. At the top of the page, choose **Discover**.
4. Under **Data and model catalog**, choose **Amazon Bedrock models**. The Amazon Bedrock in SageMaker Unified Studio playground opens at the model catalog.
5. (Optional) Choose **Group by: Modality** and select **Provider** to sort the list by model provider.
6. Choose a model to get information about the model.
7. If you're ready to build with the app, choose **Action** and select the appropriate action. You can also choose an action from the model tile on the model catalog page.
8. Choose **Amazon Bedrock model catalog** to go back to the model catalog page.

What is a prompt?

A prompt is the input that you send to a model in order for it to generate a response. For example, you could send the following user prompt to a model: **What is Avebury stone circle?**

This prompt would likely generate a response similar to the following:

Avebury stone circle is a Neolithic monument located in Wiltshire, England. It consists of a massive circular bank and ditch, with a large outer circle of standing stones that originally numbered around 100.

Some models support *multimodal* prompts, which are prompts that support different types of media input, such as text, images, or video. For example, you could send an image to a model and ask it to describe what the image contains. Not all models support multimodal prompts and modality support varies by model. For information on how to best create prompts for a specific model, see [Prompt engineering guides](#).

In addition to user prompts, Amazon Bedrock in SageMaker Unified Studio also supports inference parameters and system instructions, which allow you to customize and influence model behavior. The following sections provide information and guidance on how to use inference parameters and system prompts.

Topics

- [Inference parameters](#)
- [System instructions](#)
- [Prompt engineering guides](#)

Inference parameters

Inference parameters are values that you can adjust to influence how a model generates a response to a prompt. For example, in the chat agent app you create in [Build a chat agent app with Amazon Bedrock](#), you can use inference parameters to adjust the randomness and diversity of the songs that the model generates for a playlist.

You can apply inference parameters to models you use in the [Amazon Bedrock playgrounds](#), [chat agent apps](#), and [flow apps](#).

Randomness and diversity

For any given sequence, a model determines a probability distribution of options for the next token in the sequence. To generate each token in an output, the model samples from this distribution. Randomness and diversity refer to the amount of variation in a model's response. You can control these factors by limiting or adjusting the distribution. Foundation models typically support the following parameters to control randomness and diversity in the response.

- **Temperature**— Affects the shape of the probability distribution for the predicted output and influences the likelihood of the model selecting lower-probability outputs.
 - Choose a lower value to influence the model to select higher-probability outputs.
 - Choose a higher value to influence the model to select lower-probability outputs.

In technical terms, the temperature modulates the probability mass function for the next token. A lower temperature steepens the function and leads to more deterministic responses, and a higher temperature flattens the function and leads to more random responses.

- **Top K** – The number of most-likely candidates that the model considers for the next token.

- Choose a lower value to decrease the size of the pool and limit the options to more likely outputs.
- Choose a higher value to increase the size of the pool and allow the model to consider less likely outputs.

For example, if you choose a value of 50 for Top K, the model selects from 50 of the most probable tokens that could be next in the sequence.

- **Top P** – The percentage of most-likely candidates that the model considers for the next token.
 - Choose a lower value to decrease the size of the pool and limit the options to more likely outputs.
 - Choose a higher value to increase the size of the pool and allow the model to consider less likely outputs.

In technical terms, the model computes the cumulative probability distribution for the set of responses and considers only the top P% of the distribution.

For example, if you choose a value of 0.8 for Top P, the model selects from the top 80% of the probability distribution of tokens that could be next in the sequence.

The following table summarizes the effects of these parameters.

Parameter	Effect of lower value	Effect of higher value
Temperature	Increase likelihood of higher-probability tokens	Increase likelihood of lower-probability tokens
	Decrease likelihood of lower-probability tokens	Decrease likelihood of higher-probability tokens
Top K	Remove lower-probability tokens	Allow lower-probability tokens
Top P	Remove lower-probability tokens	Allow lower-probability tokens

As an example to understand these parameters, consider the example prompt **I hear the hoof beats of** ". Let's say that the model determines the following three words to be candidates for the next token. The model also assigns a probability for each word.

```
{  
  "horses": 0.7,  
  "zebras": 0.2,  
  "unicorns": 0.1  
}
```

- If you set a high **temperature**, the probability distribution is flattened and the probabilities become less different, which would increase the probability of choosing "unicorns" and decrease the probability of choosing "horses".
- If you set **Top K** as 2, the model only considers the top 2 most likely candidates: "horses" and "zebras."
- If you set **Top P** as 0.7, the model only considers "horses" because it is the only candidate that lies in the top 70% of the probability distribution. If you set **Top P** as 0.9, the model considers "horses" and "zebras" as they lie in the top 90% of probability distribution.

System instructions

A system instruction an overarching initial guideline that defines how a model should behave in future interactions. System instructions provide context to the model about the task it should perform or the persona it should adopt during the conversation.

For example, you could use a system instruction to specify that the model should behave as an app that creates playlists for a radio station that plays rock and pop music. You can then use the model to create playlists of rock and pop songs based on different themes, such as songs that are related by artist

You can apply system instructions to models you use in the [Amazon Bedrock playgrounds](#), [chat agent apps](#), and [flow apps](#).

Prompt engineering guides

Amazon Bedrock in SageMaker Unified Studio provides models from a variety of model providers. Each provider provides guidance on how to best create prompt for their models.

- **Amazon Nova user guide:** <https://docs.aws.amazon.com/nova/latest/userguide/what-is-nova.html>
- **Anthropic Claude model prompt guide:** <https://docs.anthropic.com/claude/docs>
- **Anthropic Claude prompt engineering resources:** <https://docs.anthropic.com/claude/docs/guide-to-anthropic-prompt-engineering-resources>
- **Cohere prompt guide:** <https://txt.cohere.com/how-to-train-your-pet-llm-prompt-engineering>
- **AI21 Labs Jurassic model prompt guide:** <https://docs.ai21.com/docs/prompt-engineering>
- **Meta Llama 2 prompt guide:** <https://ai.meta.com/llama/get-started/#prompting>
- **Stability documentation:** <https://platform.stability.ai/docs/getting-started>
- **Mistral AI prompt guide:** https://docs.mistral.ai/guides/prompting_capabilities/

For general guidelines about creating prompts with Amazon Bedrock, see [General guidelines for Amazon Bedrock LLM users](#).

Experiment with the Amazon Bedrock playgrounds

An Amazon Bedrock in SageMaker Unified Studio playground lets you experiment with Amazon Bedrock foundation models, so that you can choose the right model for your use case. You can also experiment with chat agent apps that others share with you. Amazon Bedrock in SageMaker Unified Studio provides the following playgrounds:

- **Chat playground** – Chat with an Amazon Bedrock model by sending prompts to the model and answering the response that the model generates. You can also experiment with chat agent apps that are shared with you.
- **Image and video playground** – Generate images and videos with a model. You can use prompts, images, and videos to describe the content you want to generate.

Each playground lets you choose a model and experiment with settings, such as the [inference parameters](#) that affect the output that the model generates. To help you experiment, you can compare the output of multiple models and chat agent apps.

If you want to know more about a model, use the model catalog to find information such as supported use cases and model attributes. For more information, see [Find serverless models with the Amazon Bedrock model catalog](#).

Warning

Generative AI may give inaccurate responses. Avoid sharing sensitive information. Chats may be visible to others in your organization.

Other Amazon Bedrock in SageMaker Unified Studio users can share apps and prompts so that you can experiment with them in a playground. For more information, see [Access shared generative AI assets in an Amazon Bedrock playground](#).

After you familiarize yourself with a model in a playground, you can try creating your own Amazon Bedrock in SageMaker Unified Studio app, such as a [chat agent app](#).

Topics

- [Chat with a model in the Amazon Bedrock chat playground](#)
- [Chat with an app in the Amazon Bedrock chat playground](#)
- [Generate an image with the Amazon Bedrock image and video playground](#)
- [Generate a video with the Amazon Bedrock image and video playground](#)
- [Access shared generative AI assets in an Amazon Bedrock playground](#)

Chat with a model in the Amazon Bedrock chat playground

The Amazon Bedrock in SageMaker Unified Studio chat playground allows you chat with an Amazon Bedrock model and try chat agent apps that are [shared](#) to you. A chat provides a back-and-forth, dialogue-like interaction between you and an Amazon Bedrock model. The model is able to retain context during a chat allowing for coherent and relevant responses from the model. You chat with a model by sending a prompt to the model and by receiving the response that the model generates. You continue the chat by sending further prompts.

If a model supports multimodal prompts, you can send prompts that contain text and images. A chat can contain multiple text and image prompts. After you finish a chat, you can reset the playground to begin a new chat. Amazon Bedrock models support differing modalities. For more information, see [Supported foundation models in Amazon Bedrock](#).

The maximum image file size is 5MB. You can upload images that are in JPG, PNG, GIF, and WebP format.

When you run a prompt in the chat playground, you get the following information about the request:

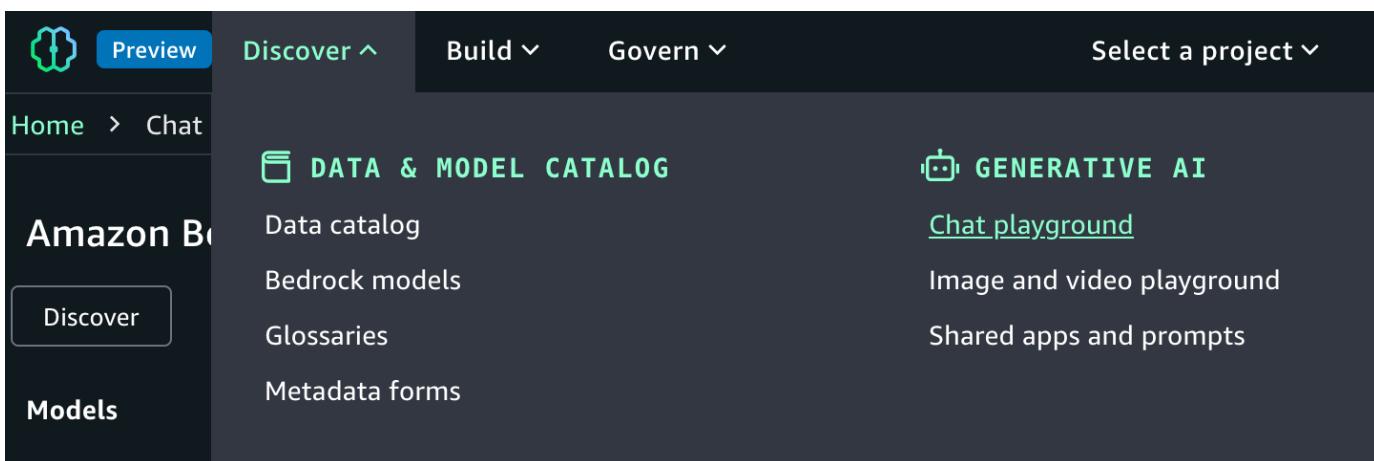
- **Input tokens** — The number of input tokens used by the foundation model during inference.
- **Output tokens** — The number of tokens generated in a response by the foundation model.
- **Latency** — The amount of time the foundation model uses to generate each token in a sequence, based on the [on-demand](#) throughput.

The [chat playground](#) provides quick start prompts that illustrate the kinds of prompt that you can send to a model.

Optionally, you can compare the outputs from up to 3 shared apps and models. You can make configuration changes for models, such as [inference parameters](#) and [system instructions](#) and compare the results. You can't make configuration changes for shared apps.

To chat with a model

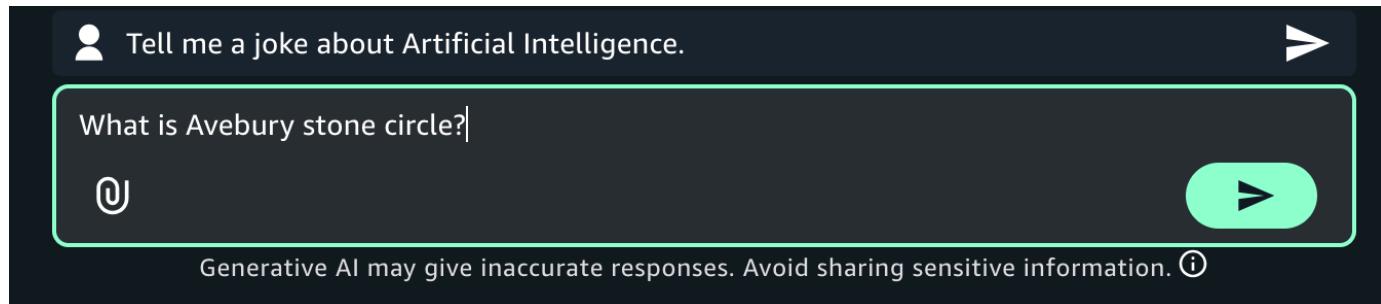
1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. At the top of the page, choose the **Discover**.
4. In the **Generative AI** section, choose **Chat playground** to open the chat playground.



5. In **Type** select **Model** and then select a model to use in **Model**. For full information about the model, choose **View full model details** in the information panel. For more information, see [Find serverless models with the Amazon Bedrock model catalog](#). If you don't have access to

an appropriate model, contact your administrator. Different models might not support all features.

6. In the **Enter prompt** text box, enter **What is Avebury stone circle?**
7. (Optional) If the model you chose is a reasoning model, you can choose **Reason** to have the model include its reasoning in the response. For more information, see [Enhance model responses with model reasoning](#) in the *Amazon Bedrock user guide*.
8. Press Enter on your keyboard, or choose the run button, to send the prompt to the model. Amazon Bedrock in SageMaker Unified Studio shows the response from the model in the playground.

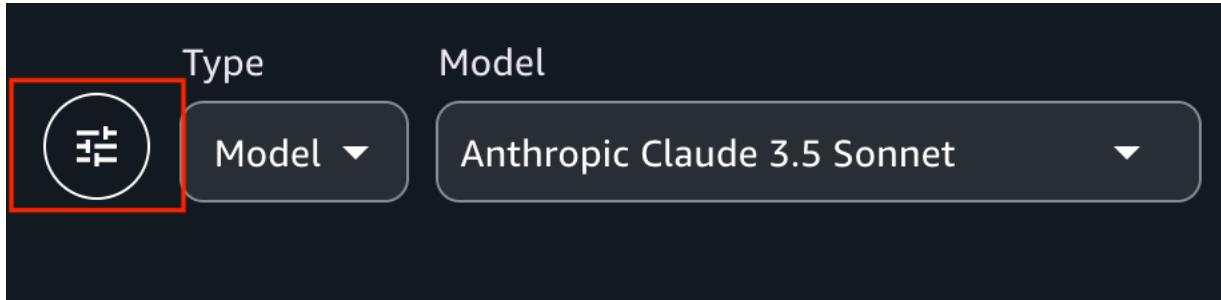


9. Continue the chat by entering the prompt **Is there a museum there?** and pressing Enter.

The response shows how the model uses the previous prompt as context for generating its next response.

10. Choose **Reset** to start a new chat with the model.
11. Influence the model response by doing the following:

- a. Enter and run a prompt. Note the response from the model.
- b. Choose the configurations menu to open the **Configurations** pane.



- c. Influence the model response by making [inference parameters](#) changes.
- d. (Optional) In **System instructions**, enter any overarching system instructions that you want the model to apply for future interactions.
- e. Run the prompt again and compare the response with the previous response.

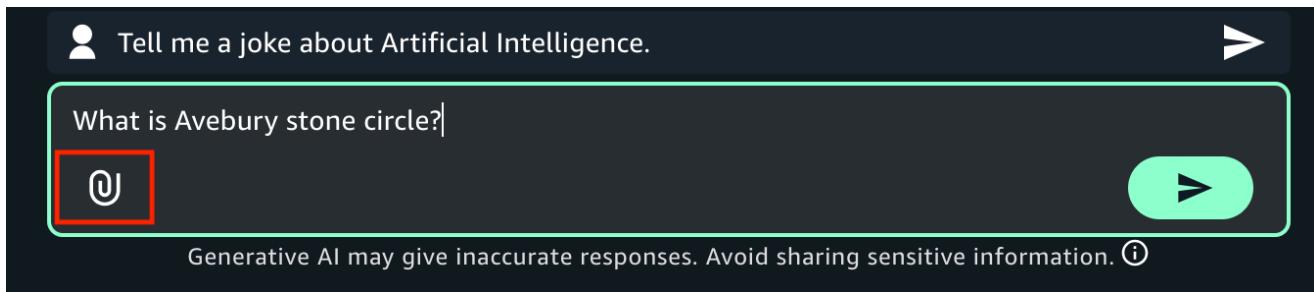
12. Choose **Reset** to start a new chat with the model.

13.

14. Try sending an image to a model by doing the following:

- For **Model**, choose a model that supports [images](#).

- Choose the attachment button at the left of the **Enter prompt** text box.



- In the open file dialog box, choose an image from your local computer.

- In the text box, next to the image that you uploaded, enter **What's in this image?**.

- Press Enter on your keyboard enter to send the prompt to the model. The response from the models describes the model or image.

15. (Optional) Try using another model and different prompts. Different models have different recommendations for creating, or engineering, prompts. For more information, see [Prompt engineering guides](#).

16. (Optional) Compare the output from multiple models, or [shared apps](#).

- In the playground, turn on **Compare mode**.

- In both panes, select the model that you want to compare. If you want to use a shared app, select **App** in **Type** and then select the app in **App**.

- Enter a prompt in the text box and run the prompt. The output from each model is shown. You can choose the copy icon to copy the prompt or model response to the clipboard.

- (Optional) Choose **View configs** to make configuration changes, such as [inference parameters](#). Choose **View chats** to return to the chat page.

- (Optional) Choose **Add chat window** to add a third window. You can compare up to 3 models or apps.

- Turn off **Compare mode** to stop comparing models.

Now that you are familiar with the explorer playground, try creating a Amazon Bedrock in SageMaker Unified Studio app next. For more information, see [Build a chat agent app with Amazon Bedrock](#).

Chat with an app in the Amazon Bedrock chat playground

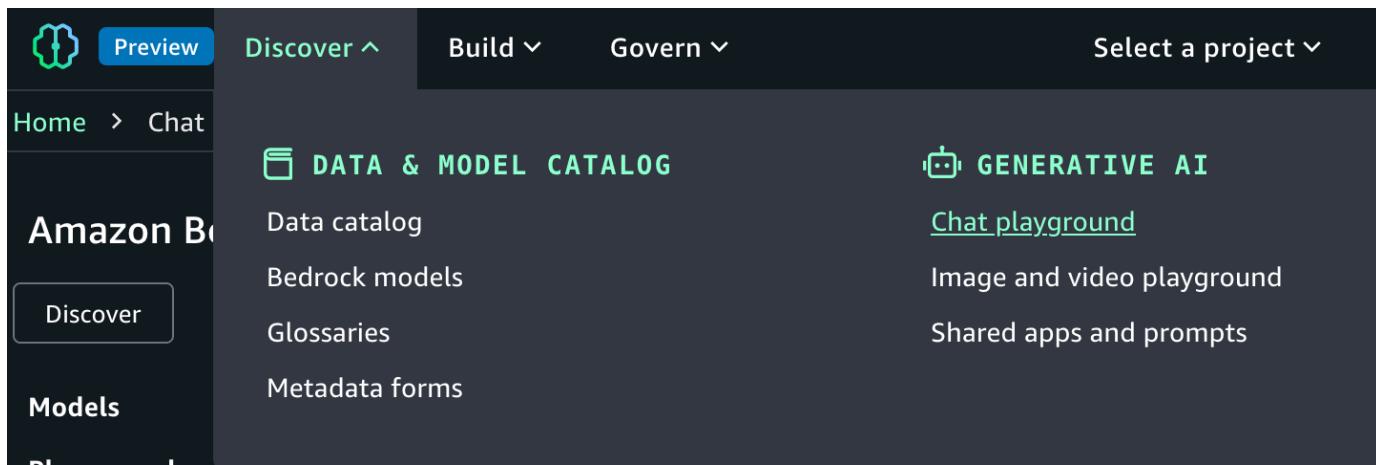
You can use the chat playground to experiment with chat agent apps that are shared to you. When you open a shared app, you can send prompts to the app and see the response. You can't make changes to the shared app.

Optionally, you can compare the outputs from to 3 shared apps and [models](#). You can view the configuration for a shared app, but you can't make configuration changes.

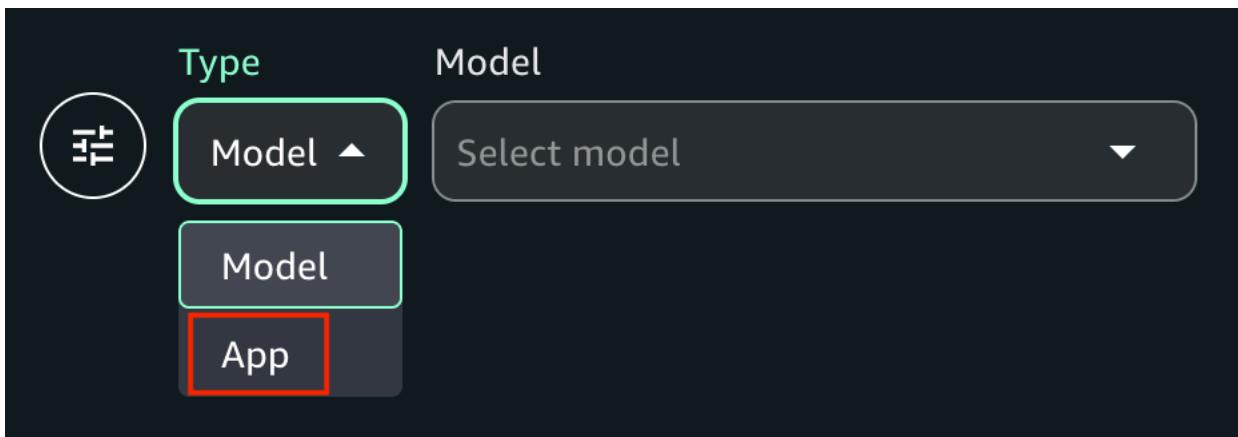
To learn how to share apps that you create, see [Share an Amazon Bedrock chat agent app](#).

To chat with a shared app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **Generative AI** section, choose **Chat playground** to open the chat playground.



5. In **Type** select **App** and then select an app to use in **App**.



6. In the **Enter prompt** text box at the bottom of the page, enter the prompt that you want to use. If the app builder changes the default text for the text box, the text is different.
7. Press Enter on your keyboard enter to send the prompt to the mode.
8. (Optional) Compare the output from multiple apps, or models.
 - a. In the playground, turn on **Compare mode**.
 - b. In both panes, select the app that you want to compare.
 - c. Enter a prompt in the text box and run the prompt.
 - d. (Optional) Choose **View configs** to view the app configurations, such as [inference parameters](#). Choose **View chats** to return to the chat page.
 - e. (Optional) Choose **Add chat window** to add a third window. You can compare up to 3 models or apps.
 - f. Turn off **Compare mode** to stop comparing models.

Generate an image with the Amazon Bedrock image and video playground

The image and video playground is an interactive environment that lets you specify actions that generate and manipulate images using natural language prompts, reference images, and suitable Amazon Bedrock models.

Actions for generating images

Within the image playground, you use an *action* to specify the image generation task that you want the model to do, such as replacing the background of an existing image. The actions that are available depends on the model you use.

- **Generate image** — [Generates a new image](#) from a prompt that you enter.
- **Generate variations** — Use a prompt to generate a [variation of an existing image](#).
- **Remove object** — [Removes an object](#) from an image you supply.
- **Replace background** — [Replaces the background](#) of an image with a new background.
- **Replace object** — [Replaces an object](#) in an image with a different object.
- **Edit image sandbox** — An [image sandbox](#) that you can use to experiment with Stable Diffusion XL models.

Some actions, such as generate variation, require a reference image that a model uses to generate a new image. An action might require you to use a mask tool to draw a bounding box around an area of the reference image, such as when you define an object that you want to remove with the remove object action.

Configuration options

You can influence how a model generates an image by configuring the following options. The configuration changes you can make depends on the action you choose.

Negative prompt

A set of words or phrases that tells the model what not to include in the image that it generates. For example, you can use the term `-lowres` to avoid generating low-resolution or blurry images.

Reference image

In certain actions, such as generate variations or replace background, you specify a reference image that the model uses to process the action.

Response image

You can specify the image dimensions, orientation, and number of images to generate.

Advanced configuration options

You can make advanced configuration changes that how the model generates images. All models image generation models support the following:

- **Prompt strength** — Prompt strength is a numerical value that determines how strongly a model should adhere to the given text prompt. A higher prompt strength means the model will try to closely follow and prioritize the text description provided in the prompt when generating the image. Lower prompt strengths allow the model more creative freedom to deviate from the prompt.
- **Seed** — A seed is numeric value that a model uses to seed a random number generator. The model uses the seed as a starting point for creating random patterns during image generation. This initial randomness influences things like the exact positioning, colors, textures, and compositions present in the image that the model generates.
- **Similarity strength** — If you use the *Generate variations* action with a Titan Image Generator G1 V1 or a Titan Image Generator G1 V2 model, you can also configure the *Similarity Strength* advanced configuration. Similarity Strength specifies how similar the generated image should be to the input image. Use a lower value to introduce more randomness into the generated image.
- **Generate step** — If you use a Stable Diffusion XL model, you can configure the *Generate step* advanced configuration. Generate step determines how many times the image is sampled. More steps can result in a more accurate result.

Generate an image

The following procedure shows you how to use a model to generate an image. You can set various configurations such as the number of images to generate and how strongly the prompt affects the generation of the image. For more information, see [Configuration options](#).

To generate an image in the image playground

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.
7. For **Action** choose the action **Generate image**.
8. In **Response image** do the following:

- a. For **Number of images** select the number of images that you want the model to generate. Not all models support changing this value.
 - b. For **Orientation**, choose the orientation (landscape or portrait) for the images that the model generates.
 - c. For **Size**, select the size, in pixels, of the images that the model generates.
9. (Optional) In **Advanced configurations**, change how the model generates images by making advanced configuration changes. For more information, see [Advanced configuration options](#).
10. In the **Enter prompt** text box, enter **Create a photo of a local classic rock band playing on an outdoor stage..** Alternatively, enter a prompt of your choosing.
11. Press Enter on your keyboard to start the action. Amazon Bedrock in SageMaker Unified Studio shows the image that the model generates in the playground.
12. (Optional) See how different configuration parameters affect image generation by repeating steps 9 - 11 with different values.

Generate a variation of an image

The following procedure shows you how to generate a variation of a reference image that you supply. You can set various configurations such as the number of images to generate and how strongly the prompt affects the generation of the image. For more information, see [Configuration options](#).

To generate a variation of an image

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.
7. For **Action** choose **Generate variations**.
8. (Optional) For **Negative prompt** enter text that describes content or concepts that you do not want the model to include in the image.

9. In **Reference image** choose **Upload image** and upload the image that you want the model to use with the action.
10. In **Response image** do the following:
 - a. For **Number of images** select the number of images that you want the model to generate. Not all models support changing this value.
 - b. For **Orientation**, choose the orientation (landscape or portrait) for the images that the model generates.
 - c. For **Size**, select the size, in pixels, of the images that the model generates.
11. (Optional) In **Advanced configurations**, change how the model generates images by making advanced configuration changes. For more information, see [Advanced configuration options](#).
12. In the **Enter prompt** text box, enter the prompt that describes the image that you want the model to generate.
13. Press Enter on your keyboard to start the action. Amazon Bedrock in SageMaker Unified Studio shows the image that the model generates in the playground.

Remove an object from an image

The following procedure shows you how to use a model to remove an object from an image that you supply. For example, you could remove an unwanted person from an image. You can set various configurations such as the number of images to generate and how strongly the prompt affects the generation of the image. For more information, see [Configuration options](#).

 **Note**

The object removal action is only available with Titan Image Generator G1 V1 and Titan Image Generator G1 V2 models.

To remove an object from an image

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. At the top of the page, choose the **Discover**.

4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.
7. For **Action** choose **Remove object**.
8. (Optional) For **Negative prompt** enter text that describes content or concepts that you don't want the model to include in the image.
9. In **Reference image** choose **Upload image** and upload the image that you want the model to use with the action.
10. (Optional) For **Negative prompt** enter text that describes content or concepts that you do not want the model to include in the image.
11. (Optional) In **Advanced configurations**, change how the model generates images by making advanced configuration changes. For more information, see [Advanced configuration options](#).
12. In the center pane, use the masking tool to draw a bounding box around the area of the image that you want the action to update. You can do the following:
 - a. Resize the bounding box by selecting a corner of the bounding box with your mouse button. Then, drag the mouse to resize the bounding box. Release the mouse button to complete resizing the bounding box.
 - b. Move the bounding box by selecting the interior of the bounding box with your mouse button. Move the bounding box to the new location and release the mouse button.
13. Press Enter on your keyboard to start the action. Amazon Bedrock in SageMaker Unified Studio shows the image that the model generates in the playground.

Replace an object in an image

The following procedure shows you how to use a model to replace an object in an image that you supply. For example, you could replace a piece of furniture in an image with a different piece of furniture. You can set various configurations such as the number of images to generate and how strongly the prompt affects the generation of the image. For more information, see [Configuration options](#).

 **Note**

The object replacement action is only available with Titan Image Generator G1 V1 and Titan Image Generator G1 V2 models.

To replace an object in an image

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.
7. For **Action** choose **Remove object**.
8. (Optional) For **Negative prompt** enter text that describes content or concepts that you do not want the model to include in the image.
9. In **Reference image** choose **Upload image** and upload the image that you want the model to use with the action.
10. (Optional) In **Advanced configurations**, change how the model generates images by making advanced configuration changes. For more information, see [Advanced configuration options](#).
11. In the center pane, use the masking tool to draw a bounding box around the area of the image that you want the action to update. You can do the following:
 - a. Resize the bounding box by selecting a corner of the bounding box with your mouse button. Then, drag the mouse to resize the bounding box. Release the mouse button to complete resizing the bounding box.
 - b. Move the bounding box by selecting the interior of the bounding box with your mouse button. Move the bounding box to the new location and release the mouse button.
12. Choose the run button on your keyboard to start the action. Amazon Bedrock in SageMaker Unified Studio shows the image that the model generates in the playground.

Replace the background for an image

The following procedure shows you how to use a model to replace the background for an image. For example, you could change the background for an image from a view of a forest to a view of city buildings. You can set various configurations such as the number of images to generate and how strongly the prompt affects the generation of the image. For more information, see [Configuration options](#).

Note

The background replacement action is only available with Titan Image Generator G1 V1 and Titan Image Generator G1 V2 models.

To replace the background for an image

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.
7. For **Action** choose **Replace background**.
8. (Optional) For **Negative prompt** enter text that describes content or concepts that you don't want the model to include in the image.
9. In **Reference image** choose **Upload image** and upload the image that you want the model to use with the action.
10. (Optional) For **Negative prompt** enter text that describes content or concepts that you do not want the model to include in the image.
11. (Optional) In **Advanced configurations**, change how the model generates images by making advanced configuration changes. For more information, see [Advanced configuration options](#).
12. In the center pane, use the masking tool to draw a bounding box around the area of the image that you want the action to preserve. The model updates the area outside of the bounding box. You can do the following:
 - a. Resize the bounding box by selecting a corner of the bounding box with your mouse button. Then, drag the mouse to resize the bounding box. Release the mouse button to complete resizing the bounding box.
 - b. Move the bounding box by selecting the interior of the bounding box with your mouse button. Move the bounding box to the new location and release the mouse button.

13. In the **Enter prompt** text box, enter a prompt that describes the background that you want the image to have.
14. Press Enter on your keyboard to start the action. Amazon Bedrock in SageMaker Unified Studio shows the image that the model generates in the playground.

Edit an image with the image sandbox

If you use the image playground with a Stable Diffusion XL model, you can use the image sandbox to make changes to a model.

 **Note**

The image sandbox action is only available with Stable Diffusion XL models.

To edit an image in the image sandbox

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.
7. For **Action** choose **Remove object**.
8. (Optional) For **Negative prompt** enter text that describes content or concepts that you do not want the model to include in the image.
9. In **Reference image** choose **Upload image** and upload the image that you want the model to use with the action.
10. (Optional) In **Advanced configurations**, change how the model generates images by making advanced configuration changes. For more information, see [Advanced configuration options](#).
11. In the center pane, use the masking tool to draw a bounding box around the area of the image that you want the action to update. You can do the following:

- a. Resize the bounding box by selecting a corner of the bounding box with your mouse button. Then, drag the mouse to resize the bounding box. Release the mouse button to complete resizing the bounding box.
 - b. Move the bounding box by selecting the interior of the bounding box with your mouse button. Move the bounding box to the new location and release the mouse button.
12. In the **Enter prompt** text box, enter the prompt that describes the edit that you want the model inside the bounding box.
13. Press Enter on your keyboard to start the action. Amazon Bedrock in SageMaker Unified Studio shows the image that the model generates in the playground.

Generate a video with the Amazon Bedrock image and video playground

The Amazon Bedrock in SageMaker Unified Studio image and video playground is where you can generate a short video with a suitable Amazon Bedrock model. To generate a video with a model, you supply a [prompt](#) that describe the video that you want to create and configuration information that influences how the model generates the video. For example, you can start with an image of a rock band and in the prompt request that you want to create an animated video of the band playing a live concert. The image and video playground also provides quick start prompts that illustrate the kinds of video that you can create.

You can download a video that you create in the image and video playground.

To create more complex videos, you can use a storyboard to connect a sequence of shots.

Your administrator sets the retention policy for videos that you upload and generate with the playground. For more information, contact your administrator.

Topics

- [Configure video generation](#)
- [Generate a video from a prompt](#)
- [Plan a video with the storyboard](#)

Configure video generation

To configure video generation, you choose a model to use and set optionally settings that influence the output of the model. Currently Amazon Bedrock in SageMaker Unified Studio supports video generation with Amazon Nova models. If you don't make any configuration changes, the playground uses the default values for the model.

Amazon Nova model settings

With Amazon Nova models, you can set the following configurations:

- **Start image** – (Optional) A reference image that model uses as a starting point for the video. The image dimensions must be 1280x720 pixels. If you supply an image with different dimensions, the playground resizes the image to 1280x720 pixels.
- **Seed** – (Optional) Initializes the random number generator used in the video generation process. Higher seed values don't correlate with any particular quality or characteristic in the output. Instead, use different seed values options to explore differing variations of output, either with or without the same prompt. Repeatedly using the same seed value and prompt creates the exact same video.

For more information, see the [Amazon Nova guide](#).

Generate a video from a prompt

The following instructions show you how to generate a video in the image and video playground. If you are using Amazon Nova 1.1, you can use the [storeyboard](#) to make more complex videos.

To generate a video

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.
6. For **Model** select a model to use.

7. (Optional) In the **Configurations** section, set parameters to influence the output of the model. Note that additional configurations can be set in **Advanced configurations**. The parameters that are available to depend on the model you use. For more information, see [Configure video generation](#).
8. In the **Enter prompt** text box, enter **Create an animated video of a local classic rock band playing on an outdoor stage..** Alternatively, enter a prompt of your choosing.
9. Press Enter on your keyboard to start generating the video. Amazon Bedrock in SageMaker Unified Studio shows the video that the model generates in the playground.
10. Choose the play button to view the video.
11. (Optional) Choose the download button to download the video to your computer.

Plan a video with the storyboard

To create more a more complex video, you can use the storyboard to plan the video that you want to create. In the storyboard you connect a sequence of shots, which the model combines to generate the video. Each shot is a prompt and an optional start frame image. Each shot the model generates is always 6 seconds in length. You can't set a specific duration for the video, but you can affect the duration by adding or removing shots in the story board. The maximum length of video that you can generate is 120 seconds.

 **Note**

You can only use the playground with Amazon Nova Reel 1.1.

To generate a video with the storyboard

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Image and video playground**.
5. If the **Configurations** pane isn't open, choose the configuration button.

6. For **Model** select a model to use.
7. (Optional) In the **Configurations** section, set parameters to influence the output of the model. Note that additional configurations can be set in **Advanced configurations**. The parameters that are available to depend on the model you use. For more information, see [Configure video generation](#).

In the storyboard, you can't set the duration for the video.

8. In the center pane, choose **Storyboard**.
9. Choose **Add shot** and do the following:
 - a. Choose **Describe what happens in this shot...** and enter the text for the prompt. You can update the prompt later, if necessary.
 - b. (Optional) Choose **Add start frame** to upload a starting image for the video.
 - c. (Optional) Choose the trash icon to remove shots than you no longer need.
10. Repeat the previous step until you have added all the shots for your video.
11. Choose the run button to start generating the video. Don't leave the page while the model generates the video.
12. When the model finishes generating the video, choose the play button to view the video.
13. (Optional) Make further edits and add shots as you need them.
14. (Optional) Download the video by right-clicking on the video and selecting **Save video as....**

Access shared generative AI assets in an Amazon Bedrock playground

Other Amazon Bedrock in SageMaker Unified Studio users can share [chat agent app](#) and [prompts](#) with you as *Shared generative AI assets*. You access assets from the **Shared apps and prompts** section in a playground. You can view the type of each asset and the projects that contains the assets.

In a playground, You can experiment with shared app and prompt assets, but you can't make changes to their configuration. If you want to make changes, you need to open the project that contains the asset. You can share chat agent apps and prompts that you create in a project. For more information, see [Share an Amazon Bedrock chat agent app](#) and [Share an Amazon Bedrock prompt version](#).

To access and use a shared asset in a playground

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. At the top of the page, choose the **Discover**.
4. In the **GENERATIVE AI** section, choose **Shared apps and prompts**.
5. In the playground, select the name of the asset that you want to use. The **Asset type** column tells the type of the asset (App or prompt).
6. Use the asset in the playground.

Build a chat agent app with Amazon Bedrock

An Amazon Bedrock in SageMaker Unified Studio chat agent app allows users to chat with an Amazon Bedrock model through a conversational interface, typically by sending text messages and receiving responses. The model analyzes the user's input, formulate an appropriate response, and carries on a dialogue with the user. You can use a chat agent apps for various purposes, such as providing customer service, answering questions, offering recommendations, or engaging in open-ended conversations on a wide range of topics. You can enhance a chat agent app app by integrating the following Amazon Bedrock features:

- **Data sources** — Enrich model responses by including context generated from an Amazon Bedrock knowledge base.
- **Guardrails** — Lets you implement safeguards for your chat agent app based on your use cases and responsible AI policies.
- **Functions** — Lets a model call a function to access a specific capability when handling a prompt.

When you first create an chat agent app, you have a working draft of the app. Changes you make to your chat agent app apply to the working draft. You iterate on your working draft until you're satisfied with the behavior of your app. At any time you can save your chat agent app.

Once you create and save chat agent app, you can do the following:

- [Share the chat agent app](#) with other users.
- [Export the chat agent app](#) for use outside of Amazon SageMaker Unified Studio.

- Use the chat agent app as an agent node in a flow app.

To use a chat agent app in a flow app you need to [deploy](#) the app. Amazon Bedrock in SageMaker Unified Studio deploys the app for you whenever you share the app with other users.

In this section you learn how to create chat agent app that uses Amazon Bedrock in SageMaker Unified Studio components such as a [data source](#) and a [guardrail](#). You also learn how to share your app with other users.

Topics

- [Create a chat agent app with Amazon Bedrock](#)
- [Deploy an Amazon Bedrock chat agent app](#)
- [Share an Amazon Bedrock chat agent app](#)

Create a chat agent app with Amazon Bedrock

In this section, you learn how create a simple Amazon Bedrock in SageMaker Unified Studio chat agent app that creates playlists for a radio station.

The app can generate playlists and get the dates and locations of upcoming shows. Later, you add the following Amazon Bedrock features.

- A guardrail to prevent songs with inappropriate song titles.
- A knowledge base that lets the app create playlists using your unique song information.
- A function that gets today's top 10 songs.

Topics

- [Step 1: Create the initial chat agent app](#)
- [Step 2: Add a guardrail to your chat agent app](#)
- [Step 3: Add a knowledge base to your chat agent app](#)
- [Step 4: Add a function call to your chat agent app](#)

Step 1: Create the initial chat agent app

In this step you create a chat agent app that generates playlists for a radio station.

To create the app, you first need to create an Amazon SageMaker Unified Studio [project](#). A project can contain multiple apps and is also where you can add the Amazon Bedrock components that you want your apps to use. Later you will add guardail, knowledge base, and function components to your app. You can share a project with other users and groups of users. For more information, see [Share an Amazon Bedrock chat agent app](#).

To help guide users of the app, you can set user interface (UI) text, such as hint text for the beginning of a chat.

In the app, you can experiment with the randomness and diversity of the response that the model returns by changing the [inference parameters](#).

While you develop your app, you work on the current draft. You can save the current draft to the app history. Later you might want to restart work from a previous draft. For more information, see [Use app history to view and restore versions of an Amazon Bedrock app](#).

Warning

Generative AI may give inaccurate responses. Avoid sharing sensitive information. Chats may be visible to others in your organization.

To create an Amazon Bedrock chat agent app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. On the Amazon SageMaker Unified Studio home page, in the **Amazon Bedrock in SageMaker Unified Studio** tile, choose **Build chat agent app** to create a new chat agent app. The **Select or create a new project to continue** dialog box opens.
4. In the **Select or create a new project to continue** dialog box, do one of the following:
 - If you want to use a new project, follow the instructions at [Create a new project](#). For the **Project profile** in step 1, choose **Generative AI application development**.
 - If you want to use an existing project, select the project that you want to use and then choose **Continue**.
5. In **Untitled App - nnnn**, enter **Radio show** as the name for your app.

6. In the **Configs** pane, do the following:
 - a. For **Model**, select a model that supports Guardrails, Data, and Function components. The description of the model tells you the components that a model supports. For full information about the model, choose **View full model details** in the information panel. For more information, see [Find serverless models with the Amazon Bedrock model catalog](#). If you don't have access to an appropriate model, contact your administrator. Different models might not support all features.
 - b. For **Enter a system instruction in Instructions for chat agent & examples**, enter **You are a chat agent app that creates 2 hour long playlists for a radio station that plays rock and pop music..**
 - c. In the **UI** section, update the user interface for the app by doing the following:
 - i. In **Hint text for empty chat** enter **Hi! I'm your radio show playlist creator..**
 - ii. In **Hint text for user input** enter **Enter a prompt that describes the playlist that you want..**
 - iii. In **Quick start prompts** choose **Edit**.
 - iv. Choose **Reset** to clear the list of quick start prompts
 - v. For **Quick-start prompt 1**, enter **Create a playlist of pop music songs..**
 - vi. (Optional). Enter quick start prompts of your choice in the remaining quick start prompt text boxes.
 - vii. Choose **Back to configs**.
7. Choose **Save** to save the current working draft of your app.
8. In the **Quick start prompts** section of the **Preview** pane, run the quick start prompt that you just created by choosing the prompt.

The app shows the prompt and the response from the model in the **Preview** pane.
9. In the prompt text box (the text should read **Enter a prompt that describes the playlist that you want**), enter **Create a playlist of songs where each song on the list is related to the next song, by musician, bands, or other connections. Be sure to explain the connection from one song to the next. .**
10. Choose the run button (or press Enter on your keyboard) to send the prompt to the model.
11. (Optional) In the **Inference parameters** section change the inference parameters. For example, include less familiar songs in the playlist by increasing the **Temperature** inference parameter.

The inference parameters you can change are *Temperature*, *Top P*, and *Top K*. Not all models support each of these inference parameters. For more information, see [Inference parameters](#).

12. (Optional) In **Stop sequences**, add one or more stop sequences for the model. A stop sequence ensures that the model stops generating text immediately after it generates text that matches the stop sequence. Stop sequences are useful for getting precise answers without unnecessary additional text. Not all models support stop sequences.
13. (Optional) In **Reasoning** select **Model reasoning** to enable model reasoning. Model reasoning is when a model uses chain of thought reasoning to take a large, complex task and break it down into smaller, simpler steps. You can only enable or disable model reasoning at the start of a new chat. You can specify the number of tokens to use, which includes both output and reasoning tokens. Not all models support model reasoning. For more information, see [Enhance model responses with model reasoning](#) in the *Amazon Bedrock user guide*.
14. (Optional) Share your app with others by following the instructions at [Share an Amazon Bedrock chat agent app](#).
15. (Optional) Deploy your app for use in a flow app by following the instructions at [Deploy an Amazon Bedrock chat agent app](#).
16. (Optional) Export your snapshot from Amazon SageMaker Unified Studio by following the instructions at [Use your app outside of Amazon SageMaker Unified Studio](#).
17. Next step: Add a guardrail to your app by following the instructions at [Step 2: Add a guardrail to your chat agent app](#).

Step 2: Add a guardrail to your chat agent app

Guardrails for Amazon Bedrock lets you implement safeguards for your Amazon Bedrock in SageMaker Unified Studio app based on your use cases and responsible AI policies. You can create multiple guardrails tailored to different use cases and apply them across multiple foundation models, providing a consistent user experience and standardizing safety controls across generative AI apps. You can configure denied topics to disallow undesirable topics and content filters to block harmful content in the prompts you send to a model and to the responses you get from a model. You can use guardrails with text-only foundation models. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).

Add a guardrail

This procedure shows you how to use a guardrail to safeguard the app you created in [Step 1: Create the initial chat agent app](#). The guardrail prevents inappropriate language in song titles and filters out unwanted music genres.

To add a guardrail to an Amazon Bedrock app

1. Open the app that you created in [Step 1: Create the initial chat agent app](#).
2. In the **Configs** pane, choose **Guardrails** and then **Create new guardrail**.
3. For **Guardrail name**, enter **prevent_unwanted_songs**.
4. For **Guardrail description**, enter **Prevents inappropriate or undesirable songs..**
5. In **Content filters** make sure **Enable content filters** is selected. For more information, see [Content filters](#).
6. In **Filter for prompts** make sure the filter for each category is set to **High**.
7. Make sure **Apply the same filters for responses** is selected.
8. In **Blocked messaging** do the following.
 - a. For **Blocked messaging for prompts**, enter **Sorry, your prompt contained inappropriate text..**
 - b. Clear **Apply the same message for blocked responses**.
 - c. For **Blocked messaging for responses**, enter **Sorry, but I can't respond with information that contains inappropriate text..**
9. Choose **Create** to create the guardrail.
10. In the **Configs** pane, in the **Guardrails** section, select the guardrail that you just created (**prevent_unwanted_songs**). It might take a minute for the guardrail to appear in the list.
11. Test the guardrail by entering **Create a list of 10 songs where each song has a swear word in the title**. In the prompt edit box.
12. Choose the run button to send the prompt to the model. The model should respond with the message **Sorry, but I can't respond with information that contains inappropriate text**.
13. Use a denied topic filter to prevent requests for music from a specific music genre. For information about denied topics, see [Denied topics](#).

To add the filter, do the following.

- a. In the **Guardrails** section of the **Configs** pane, select the guardrail and choose **Preview**.

- b. Choose **Edit** to edit the guardrail.
 - c. In **Denied topics**, choose **Add topic**.
 - d. For **Name**, enter **heavy metal**.
 - e. For **Definition for topic**, enter **Avoid mentioning songs that are from the heavy metal genre of music..**
 - f. In **Sample phrases - optional**, enter **Create a playlist of heavy metal songs.**
 - g. (Optional) Choose **Add phrase** to add other phrases.
 - h. Choose **Save**.
 - i. On the **Edit guardrail** page, choose **Update** to update the guardrail.
 - j. Test the guardrail by entering **Create a list of heavy metal songs.** in the prompt edit box.
 - k. Choose the run button to send the prompt to the model. The model should respond with the message **Sorry, your prompt contained inappropriate text.**
14. Next step: Add a data source to your app by following the instructions at [Step 3: Add a knowledge base to your chat agent app](#).

Step 3: Add a knowledge base to your chat agent app

You can use your own data into your application by adding a knowledge base to your app. Doing this allows your app to access to information that is only available to you. When your app passes a query, Amazon Bedrock in SageMaker Unified Studio generates a response that includes the query results from the knowledge base. For more information, see [Add a data source to your Amazon Bedrock app](#).

In this topic, you update the app you created in [Step 1: Create the initial chat agent app](#) to use a CSV file as a document (local file) data source for a knowledge base. The CSV file includes information about bands that don't have public metadata such as song length, or music genre. The user can use the app to create a playlist based on criteria such as song length or music genre.

To add your own data to an Amazon Bedrock app

1. Create a CSV file name *songs.csv* and fill with the following fictitious CSV data.

```
song,artist,genre,length-seconds
"Celestial Odyssey","Starry Renegades","Cosmic Rock",240
"Neon Rapture","Synthwave Siren","Synthwave Pop",300
```

```
"Wordsmith Warriors","Lyrical Legions","Lyrical Flow",180
"Nebula Shredders","Galactic Axemen","Cosmic Rock",270
"Electro Euphoria","Neon Nomads","Synthwave Pop",210
"Rhythm Renegades","Percussive Pioneers","Lyrical Flow",240
"Stardust Rift","Cosmic Crusaders","Cosmic Rock",180
"Synthwave Serenade","Electro Enchanters","Synthwave Pop",300
"Lyrical Legends","Rhyme Royale","Lyrical Flow",240
"Supernova Shredders","Amplified Ascension","Cosmic Rock",300
"Celestial Chords","Ethereal Echoes","Cosmic Rock",240
"Neon Nirvana","Synthwave Sirens","Synthwave Pop",270
"Verbal Virtuoso","Lyrical Maestros","Lyrical Flow",210
"Cosmic Collision","Stellar Insurgents","Cosmic Rock",180
"Pop Paradox","Melodic Mavericks","Synthwave Pop",240
"Flow Fusion","Verbal Virtuosos","Lyrical Flow",300
"Shredding Shadows","Crimson Crusaders","Cosmic Rock",270
"Synth Serenade","Electro Enchanters","Synthwave Pop",180
"Wordsmith Warlords","Lyrical Legionnaires","Lyrical Flow",240
"Sonic Supernova","Amplified Ascension","Cosmic Rock",210
"Celestial Symphony","Ethereal Ensemble","Cosmic Rock",300
"Electro Euphoria","Neon Nomads","Synthwave Pop",180
"Lyrical Legends","Rhyme Royale","Lyrical Flow",270
"Crimson Crescendo","Scarlet Serenaders","Cosmic Rock",240
"Euphoric Tides","Melodic Mystics","Synthwave Pop",210
"Rhythm Renegades","Percussive Pioneers","Lyrical Flow",180
"Cosmic Collision","Stellar Insurgents","Cosmic Rock",300
"Stardust Serenade","Celestial Crooners","Synthwave Pop",240
"Wordsmith Warriors","Lyrical Legions","Lyrical Flow",270
"Sonic Supernova III","Amplified Ascension","Cosmic Rock",180
```

2. Open the app that you created in [Step 1: Create the initial chat agent app](#).
3. In **Data** choose **Use Knowledge Base** and then **Create Knowledge Base**. The **Create Knowledge Base** pane is shown. If you've previously created the knowledge base, go to step 10 and select the knowledge base.
4. For **Name**, enter a name for the Knowledge Base.
5. For **Description**, enter a description for the Knowledge Base.
6. In **Add data sources**, choose **Local file**.
7. Choose **Click to upload** and upload the CSV file that you created in step 1. Alternatively, add the CSV by dragging and dropping the document from your computer.

For more information, see [Document data source](#).

8. For **Embeddings model**, choose a model for converting your data into vector embeddings.

9. Choose **Create**. It might take Amazon Bedrock in SageMaker Unified Studio a few minutes to create the knowledge base.
10. For **Select Knowledge Base**, select the Knowledge Base that you just created.
11. Test the data source by entering **Create a playlist of songs in the Lyrical Flow genre** in the prompt text box.
12. Choose the run button to send the prompt to the model. The model should respond with a playlist of songs from the Lyrical Flow genre that the CSV file contains.
13. Choose **Save** to save the app.

Step 4: Add a function call to your chat agent app

Amazon Bedrock in SageMaker Unified Studio functions let a model include information that it has no previous knowledge of in its response. For example, you can use a function to include dynamic information in a model's response such as a weather forecast, sports results, or traffic conditions. To use a function in Amazon Bedrock in SageMaker Unified Studio you add a function component to your app. For more information, see [Call functions from your Amazon Bedrock chat agent app](#).

In Amazon Bedrock in SageMaker Unified Studio, a function calls an API hosted outside of Amazon Bedrock in SageMaker Unified Studio. You either create the API yourself, or use an existing API. To create an API, you can use [Amazon API Gateway](#).

In this procedure, you add a function to the app that you created in [Step 1: Create the initial chat agent app](#) so that users can get a list of the top 10 songs played on the radio station that day.

To add a function to an Amazon Bedrock app

1. Create a HTTPS server that implements a TopSongsToday function. Make sure the function adheres to the following schema.

```
openapi: 3.0.0
info:
  title: Top Songs API
  description: API to retrieve the top 10 songs played today
  version: 1.0.0

paths:
  /top-songs:
    get:
      operationId: TopSongsToday
```

```
summary: Get the top 10 songs played today
description: >
  This endpoint returns an array of the top 10 songs played today,
  ordered by popularity. The first element in the array (index 0)
  represents the most popular song, and the last element (index 9)
  represents the 10th most popular song.
responses:
  '200':
    description: Successful response
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/TopSongs'

components:
  schemas:
    TopSongs:
      type: array
      items:
        $ref: '#/components/schemas/Song'
      description: >
        An array containing the top 10 songs played today. The first element
        (index 0) is the most popular song, and the last element (index 9)
        is the 10th most popular song.
    example:
      - title: 'Song Title 1'
        artist: 'Artist Name 1'
        album: 'Album Name 1'
      - title: 'Song Title 2'
        artist: 'Artist Name 2'
        album: 'Album Name 2'
      # ... up to 10 songs

  Song:
    type: object
    properties:
      title:
        type: string
        description: The title of the song
      artist:
        type: string
        description: The name of the artist or band
      album:
        type: string
```

```
    description: The name of the album the song is from
    required:
      - title
      - artist
      - album
```

2. Open the app that you created in [Step 1: Create the initial chat agent app](#).
3. In **Models**, choose a model that supports functions. If you don't have access to an appropriate model, contact your administrator.
4. In **Functions**, choose **Create new function**.
5. In the **Create function** pane, do the following.
 - a. For in **Function name**, enter **Top_ten_songs_today**.
 - b. For **Function description (optional)**, enter **Today's top 10 songs..**
 - c. For **Function schema**, enter the OpenAPI schema from step one.
 - d. Choose **Validate schema** to validate the schema.
 - e. In **Authentication method** choose the authentication method for your HTTP server. For more information, see [the section called "Authentication methods"](#).
 - f. In **API servers**, enter the URL for your server in **Server URL**. This value is autopopulated if the server URL is in the schema.
 - g. Choose **Create** to create your function. It might take a few minutes to create the function.
6. For **Enter a system instruction**, update the system instruction so that it describes the function. Use the following text: **You are an app that creates 2 hour long playlists for a radio station that plays rock and pop music. The function Top_ten_songs_today gets the most popular song played on the radio station..**
7. Test the function by doing the following.
 - a. Enter **What are today's top 10 songs?** in the prompt edit box.
 - b. Choose the run button to send the prompt to the model. The model should respond with the list of today's top 10 songs.

Deploy an Amazon Bedrock chat agent app

Deployment of a chat agent app happens for the following reasons:

- You want to use a chat agent app as an [agent node](#) within a flow app. Before you can do so, you must first deploy the chat agent app.
- You want to [share](#) a chat agent app with other users. When you share the app, Amazon Bedrock deploys the chat agent app for you.

During deployment Amazon Bedrock creates, or reuses, an *alias* for the chat agent app and creates a new *version* of the chat agent app.

Aliases and versions

An alias is associated with a specific version of a chat agent app. When you use a chat agent app as an agent node in a flow app, you configure the agent node in the flow app with the alias of the chat agent app. When you share an app, the recipient can access the version of the app that is associated with the alias. You can create up to 8 aliases for a chat agent app.

A version is a snapshot of a chat agent app at the time you deploy the app. Amazon Bedrock creates a new version of your chat agent app each time you deploy the app. Amazon Bedrock creates versions in numerical order, starting from 1.

With aliases, you can switch efficiently between [different versions](#) of a chat agent app, without having to update the flow apps that use the chat agent app. For example, if you discover an issue with the current version of your chat agent app, you can quickly update agent nodes that use the current version to a previous version.

The following procedure shows you how to deploy a chat agent app. After deploying the app, you can then use the app as a [node agent](#) in a flow app. You don't need to deploy a chat agent app before sharing the app, as Amazon Bedrock deploys the app for you. For more information, see [Deploy an Amazon Bedrock chat agent app](#).

To deploy a chat agent app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.

- b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
 5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
 6. Open the app that you want to deploy.
 7. Choose **Deploy**.
 8. In **App description** enter a short description for the app. Make sure that that the description lets users understand the purpose of the app.
 9. Assign an alias by doing one of the following:
 - Choose **Create a new alias** to create a new alias. Then enter an name and optional description for the alias.
 - Choose **Select an existing alias** to use an existing alias. Then select the existing alias that you want to use.
 10. Choose **Deploy** to deploy the app. It might take a few seconds to deploy the app.
 11. To use the deployed app, select **Working draft** in the configuration pane and then select the name of the alias that you used to deploy the app.
 12. Enter a prompt to try your deployed app.

Modify the version of an Amazon Bedrock chat agent app

You can change the version of a chat agent app that an agent node in a flow app uses, or the version of a chat agent app that is shared with other users. To change the version, you modify the alias for the chat agent app to reference the new version. After updating the alias, you don't need to update the flow app or shares of the chat agent app for them to use the new version.

The following procedure shows you how to change the version of a chat agent app that an alias for the app references.

To modify the version that an alias references

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).

3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. Open the app that you want to use.
7. Choose the selector on the **Deploy** button and select **View aliases**. The **View and manage aliases** pane opens.
8. For the alias that you want modify, choose **Edit**.
9. In the **Edit alias** pane, select the version that you want the alias to use in **Select version to associate with this alias**.
10. (Optional) Update the name and description for the alias.
11. Choose **Save** to save your changes.

Share an Amazon Bedrock chat agent app

A snapshot of an Amazon Bedrock in SageMaker Unified Studio chat agent app is a point-in-time capture of the app's state, including its code, configuration, and any associated data.

You can share a snapshot with all members of your Amazon SageMaker Unified Studio domain, or with specific users or groups in your Amazon SageMaker Unified Studio domain. When you first share a snapshot, you get a share link to the snapshot that you can send to users. If you share the snapshot with all users, Amazon SageMaker Unified Studio grants permission to a user, when they first open the share link. Amazon SageMaker Unified Studio also adds the snapshot to the user's shared assets list. If you share the snapshot with specific users and groups, the snapshot is immediately available in their shared assets list. They can also use the share link to access the snapshot. By default, sharing a snapshot is restricted to only those users or groups that you select.

When you share a chat agent app, Amazon SageMaker Unified Studio also publishes the chat agent app to the Amazon SageMaker AI Catalog.

When sharing occurs, Amazon Bedrock deploys the snapshot of the chat agent app. When you first share a snapshot, Amazon Bedrock creates a new alias for the chat agent app and a new version of

the chat agent app that represents the snapshot. On subsequent shares, Amazon Bedrock creates a new version of the app, and associates the alias with the new version. If necessary, You can change the version that is associated with an alias. For more information, see [Modify the version of an Amazon Bedrock chat agent app](#).

To share a chat agent app snapshot

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. Open the chat agent app that you want to share.
7. Choose **Share**.
8. In **App description** enter a short description for the chat agent app. Make sure that that the description lets users understand the purpose of the chat agent app.
9. Do one of the following:
 - If you want to share the chat agent app snapshot with all members of your Amazon SageMaker Unified Studio domain, select turn on **Grant access with link**.
 - If you want to share the chat agent app snapshot with specific Amazon SageMaker Unified Studio domain users or groups, do the following in **Share with specific users or groups**:
 1. For **Member type** choose **Individual user or Group**, depending on the type of member that you want share the chat agent app with.
 2. Search for the users or groups that you want to share the chat agent app with by entering the user name or group in the **Search by alias to invite members** text box.
 3. In the drop down list, select the matching user name or group that want to share the chat agent app with.

4. Choose **Add** to add the user or group.
10. Choose **Share** to share the chat agent app.
11. When the success message appears, choose **Copy link** and send the link to the users that you are sharing the chat agent app snapshot with. If **Grant access with link** is off, the link only works for users that you have explicitly granted access to the chat agent app.

As the chat agent app creator, you, and other members of the chat agent app project, can make changes to the chat agent app and share a fresh snapshot of the chat agent app. Only the latest snapshot is available to users. Users that you share a snapshot to can't make any changes to the snapshot.

To change who you share the chat agent app with, open the chat agent app, choose **Share** and make your changes. Choose **Done** to complete the changes. If you are sharing the snapshot with all users, turning off **Grant access with link** restricts access to users that you have specifically share the snapshot with.

You can't stop sharing a snapshot without deleting the chat agent app. If you delete the chat agent app, the snapshot is no longer shared and is removed from the Amazon SageMaker AI Catalog. If you want to deny access to everyone, without deleting the chat agent app, edit the snapshot and remove all users and groups. If you shared the snapshot with all users, turn off **Grant access with link**. Note that Amazon SageMaker Unified Studio doesn't remove the snapshot from the Amazon SageMaker AI Catalog.

If you need the share link later, share the chat agent app again and copy the share link. You can also change the users that you share with the chat agent app with.

To see which chat agent apps you have shared, Open the app project, choose **Asset gallery** and then **My apps**. Check the **Share status** column for the chat agent app.

Build a flow app with Amazon Bedrock

A flow app let you link prompts, supported foundational models, and other units of work, such as an Amazon Bedrock knowledge base, together and create generative AI workflows for end-to-end solutions. For example, you could create flow apps to do the following.

- **Create a playlist of music** – Create a flow connecting a prompt node and knowledge base node. Provide the following prompt to generate a playlist: **Create a playlist**. After processing the

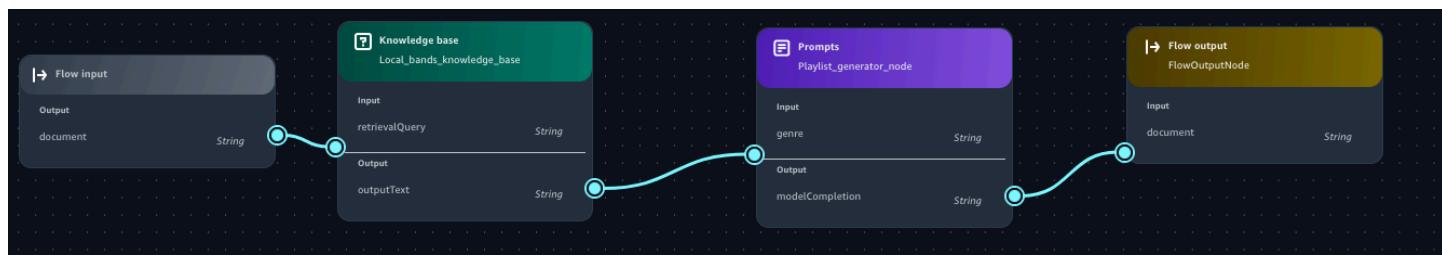
prompt, the flow queries a knowledge base to look up information about local bands, such as the length of songs and genre of music. The flow then generates a playlist based the information in the knowledge base.

- **Troubleshoot using the error message and the ID of the resource that is causing the error**

– The flow looks up the possible causes of the error from a documentation knowledge base, pulls system logs and other relevant information about the resource, and updates the faulty configurations and values for the resource.

In this section you create a flow app that generates a playlist of music from a knowledge base of songs by fictional local bands.

To create a flow app, you use the *flow builder* which is a tool in Amazon Bedrock in SageMaker Unified Studio to build and edit flow apps through a visual interface. You use the visual interface to drag and drop nodes onto the interface and configure inputs and outputs for these nodes to define your flow.



In your flow you can apply logical conditions to direct the output from a node to different destinations. You can then run the flow within Amazon Bedrock in SageMaker Unified Studio and view the output.

The following list introduces you to the basic elements of a flow.

- **Flow** – A flow is a construct consisting of a name, description, permissions, a collection of nodes, and connections between nodes. When you run a flow, the input to the flow is sent through each node of the flow until the flow emits the final output from an output node.
- **Node** – A node is a step inside a flow. For each node, you configure its name, description, input, output, and any additional configurations. The configuration of a node differs based on its type.

For information about the types of nodes that Amazon Bedrock in SageMaker Unified Studio supports, see [Flow nodes available in Amazon Bedrock](#).

- **Connection** – There are two types of connections used in flow apps:

- A **data connection** is drawn between the output of one node (the *source node*) and the input of another node (the *target node*) and sends data from an upstream node to a downstream node. In the flow builder, data connections are solid lines.
- A **conditional connection** is drawn between a condition in a condition node and a downstream node and sends data from the node that precedes the condition node to a downstream node if the condition is fulfilled. In the flow builder, conditional connections are dotted lines.
- **Expressions** – An expression defines how to extract an input from the whole input entering a node. To learn how to write expressions, see [Define inputs with expressions](#).

If you want to use your flow app outside of Amazon SageMaker Unified Studio, you can export and deploy the app to an AWS account. For more information, see [Use your app outside of Amazon SageMaker Unified Studio](#).

Warning

Generative AI may give inaccurate responses. Avoid sharing sensitive information. Chats may be visible to others in your organization.

Topics

- [Create a flow app with Amazon Bedrock](#)
- [Define inputs with expressions](#)
- [Use logic nodes to control flow](#)
- [Use a chat agent app in a flow app](#)
- [Flow nodes available in Amazon Bedrock](#)

Create a flow app with Amazon Bedrock

In this section you first build a flow app that generates a playlist of music from an Amazon Bedrock knowledge base of songs by fictional local bands. Next, you use [Reusable prompts](#) to add a prompt that can customizes the playlist for different genres of music.

Topics

- [Step 1: Create an initial flow app](#)

- [Step 2: Add a Knowledge Base to your flow app](#)
- [Step 3: Add a prompt to your flow app](#)
- [Step 4: Add a condition to your flow app](#)

Step 1: Create an initial flow app

In this procedure you create an initial flow app which has an [Flow input node](#) and a [Flow output node](#).

A flow contains only one flow input node which is where the flow begins. The flow input node takes your input and passes it to the next node in a data type of your choice (String, Number, Boolean, Object and Array). In these procedures, the input to the flow is a String. To learn more about using different data types in a flow, see [Define inputs with expressions](#).

A flow output node extracts the input data from the previous node, based on the defined expression, and outputs the data. A flow can have multiple flow output nodes if there are multiple branches in the flow.

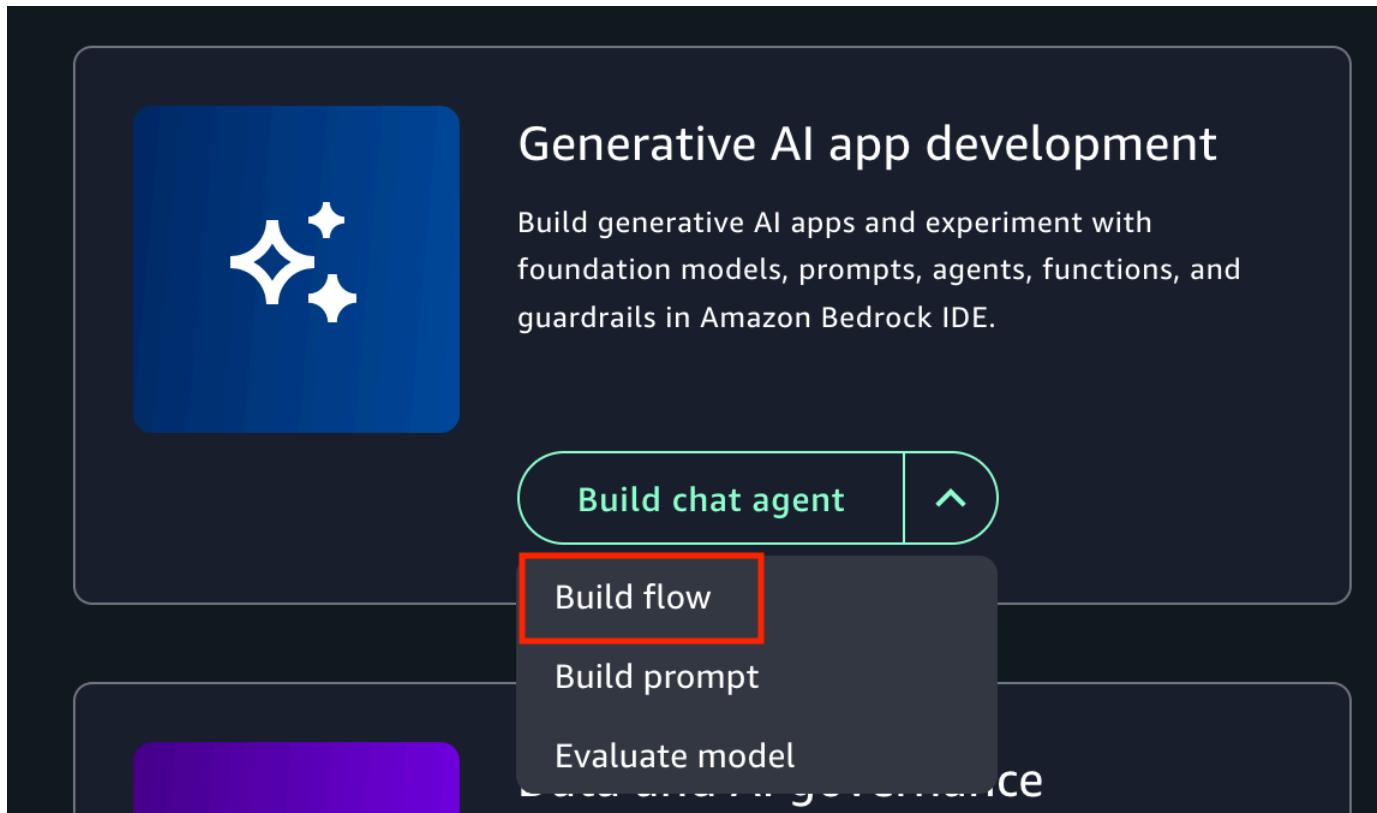
After completing the procedure, the flow app is empty, other than the flow input and flow output nodes. In the next step you add Knowledge Base as a data source and run the flow app for the first time.

While you develop your app, you work on the current draft. You can save the current draft to the app history. Later you might want to restart work from a previous draft. For more information, see [Use app history to view and restore versions of an Amazon Bedrock app](#).

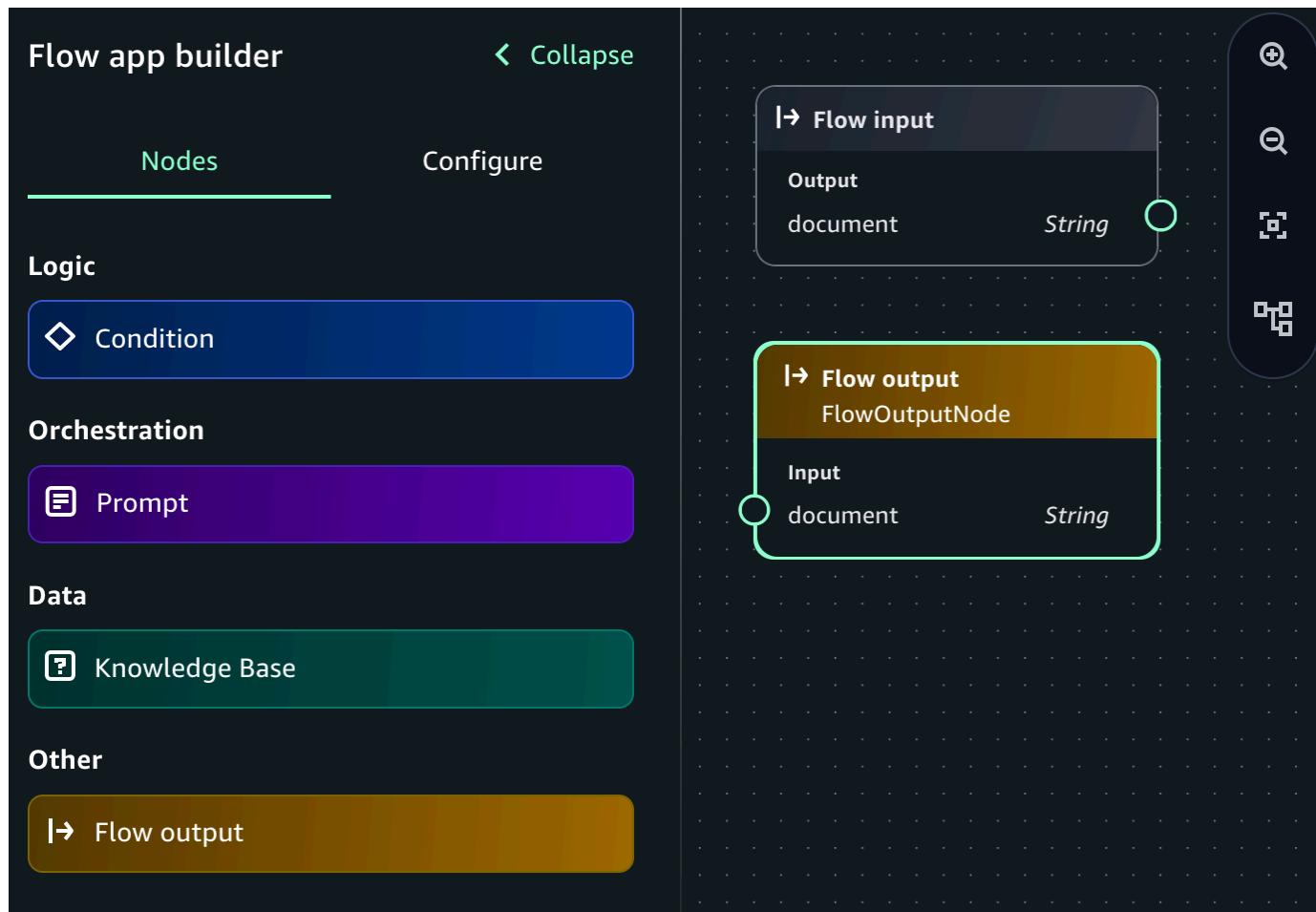
To create an initial flow app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. On the Amazon SageMaker Unified Studio home page, navigate to the **Amazon Bedrock in SageMaker Unified Studio** tile.

For the **Build chat agent app** button dropdown, select **Build flow**.



4. In the **Select or create a new project to continue** dialog box, do one of the following:
 - If you want to use a new project, follow the instructions at [Create a new project](#). For the **Project profile** in step 1, choose **Generative AI application development**.
 - If you want to use an existing project, select the project that you want to use and then choose **Continue**.
5. In the flow builder, choose the flow name (**Untitled flow-nnnn**) and enter **Local bands** as the name for the flow.
6. In the **flow app builder** pane, select the **Nodes** tab. The center pane displays a **Flow input** node and a **Flow output** node. These are the input and output nodes for your flow. The circles on the nodes are connection points. In the next procedure, you use the connection points to connect a Knowledge Base node to the Flow input node and the Flow output node.



7. Next step: [Step 2: Add a Knowledge Base to your flow app.](#)

Step 2: Add a Knowledge Base to your flow app

In this procedure, you add a [Knowledge Base](#) node as a data source to the flow that you created in [Step 1: Create an initial flow app](#). The Knowledge Base you add is Comma Separated Values (CSV) file containing a list of fictitious songs and artists. The list includes the duration (seconds) and genre of each song. For more information about Knowledge Bases, see [Add a data source to your Amazon Bedrock app](#).

During the procedure, you make connections from the Flow input node to the Knowledge Base node and from the Knowledge Base node to the Flow output node. At some point, you might need to delete a node or remove a node connection. To delete a node, select the node that you want to delete and press the Delete button. To remove a connection, choose the connection that you want to delete and then press the delete button.

When you run the flow with the input **Create a playlist**, the app creates a playlist using songs only from the Knowledge Base.

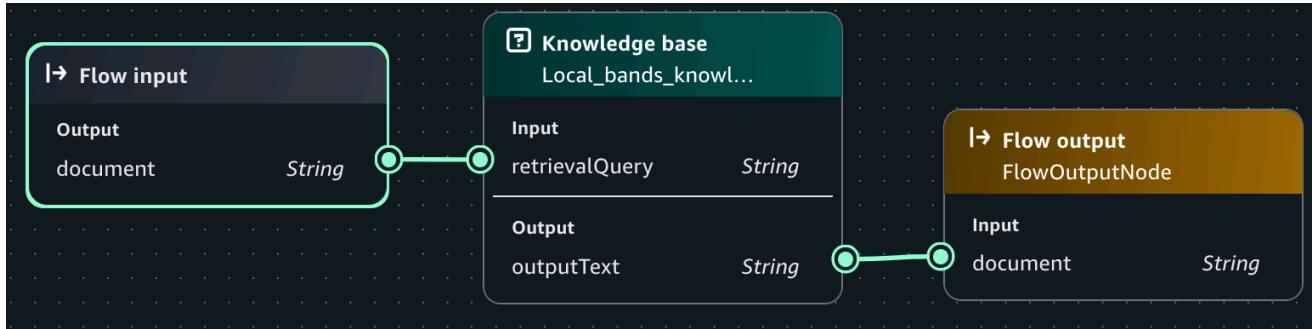
To create the flow with a Knowledge Base

1. Create a CSV file name *songs.csv* and fill with the following fictitious CSV data. This is the data source for your Knowledge Base. Save the CSV file to your local computer.

```
song,artist,genre,length-seconds
"Celestial Odyssey","Starry Renegades","Cosmic Rock",240
"Neon Rapture","Synthwave Siren","Synthwave Pop",300
"Wordsmith Warriors","Lyrical Legions","Lyrical Flow",180
"Nebula Shredders","Galactic Axemen","Cosmic Rock",270
"Electro Euphoria","Neon Nomads","Synthwave Pop",210
"Rhythm Renegades","Percussive Pioneers","Lyrical Flow",240
"Stardust Rift","Cosmic Crusaders","Cosmic Rock",180
"Synthwave Serenade","Electro Enchanters","Synthwave Pop",300
"Lyrical Legends","Rhyme Royale","Lyrical Flow",240
"Supernova Shredders","Amplified Ascension","Cosmic Rock",300
"Celestial Chords","Ethereal Echoes","Cosmic Rock",240
"Neon Nirvana","Synthwave Sirens","Synthwave Pop",270
"Verbal Virtuoso","Lyrical Maestros","Lyrical Flow",210
"Cosmic Collision","Stellar Insurgents","Cosmic Rock",180
"Pop Paradox","Melodic Mavericks","Synthwave Pop",240
"Flow Fusion","Verbal Virtuosos","Lyrical Flow",300
"Shredding Shadows","Crimson Crusaders","Cosmic Rock",270
"Synth Serenade","Electro Enchanters","Synthwave Pop",180
"Wordsmith Warlords","Lyrical Legionnaires","Lyrical Flow",240
"Sonic Supernova","Amplified Ascension","Cosmic Rock",210
"Celestial Symphony","Ethereal Ensemble","Cosmic Rock",300
"Electro Euphoria","Neon Nomads","Synthwave Pop",180
"Lyrical Legends","Rhyme Royale","Lyrical Flow",270
"Crimson Crescendo","Scarlet Serenaders","Cosmic Rock",240
"Euphoric Tides","Melodic Mystics","Synthwave Pop",210
"Rhythm Renegades","Percussive Pioneers","Lyrical Flow",180
"Cosmic Collision","Stellar Insurgents","Cosmic Rock",300
"Stardust Serenade","Celestial Crooners","Synthwave Pop",240
"Wordsmith Warriors","Lyrical Legions","Lyrical Flow",270
"Sonic Supernova III","Amplified Ascension","Cosmic Rock",180
```

2. Open the flow app that you created in [Step 1: Create an initial flow app](#).
3. Add and configure a Knowledge Base node by doing the following:

- a. In the **flow app builder** pane, select the **Nodes** tab.
- b. From the **Data** section, drag a **Knowledge Base** node onto the flow builder canvas.
- c. The circles on the nodes are connection points. Using your mouse, click on the circle for the **Flow input** node and draw a line to the circle on **Input** section of the Knowledge Base node that you just added.
- d. Connect the **Output** of the Knowledge Base node in your flow with the **Input** of the **Flow output** node.



- e. Select the Knowledge Base node that you just added.
- f. In the **flow builder** pane, choose the **Configure** tab and do the following:
 - i. In **Node name** enter **Local_bands_knowledge_base**.
 - ii. In **Knowledge Base Details**, choose **Create new Knowledge Base** to open the **Create Knowledge Base** pane.
 - iii. For **Knowledge Base name**, enter **Local-bands**.
 - iv. For **Knowledge Base description**, enter **Songs by local bands. Includes song, artist, genre, and song length (in seconds)**.
 - v. In **Add data sources**, choose **Local file**.
 - vi. Choose **Click to upload** and upload the CSV file (`songs.csv`) that you created in step 1. Alternatively, add your source document by dragging and dropping the CSV from your computer.
 - vii. For **Parsing** leave as **Default parsing**.
 - viii. For **Embeddings model**, choose a model for converting your data into vector embeddings.
 - ix. For **Vector store**, choose **OpenSearch Serverless**.
 - x. Choose **Create** to create the Knowledge Base. It might take a few minutes to create the Knowledge Base.

- g. Back in the **flow builder** pane, in **Select Knowledge Base**, select the Knowledge Base that you just created (Local-bands).
 - h. In **Select response generation model**, select the model that you want the Knowledge Base to generate responses with.
 - i. (Optional) In **Select guardrail** select an existing guardrail or create a new guardrail. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).
4. Choose **Save** to save the app.
 5. Test your prompt by doing the following:
 - a. On right side of the page, choose < to open the test pane.
 - b. Enter **Create a playlist** in the prompt **Text** box.
 - c. Press Enter on the keyboard or choose the run button to test the prompt.
 - d. If necessary, make changes to your flow. If you are satisfied with the flow, choose **Save**.
 6. Next step: [Step 3: Add a prompt to your flow app](#).

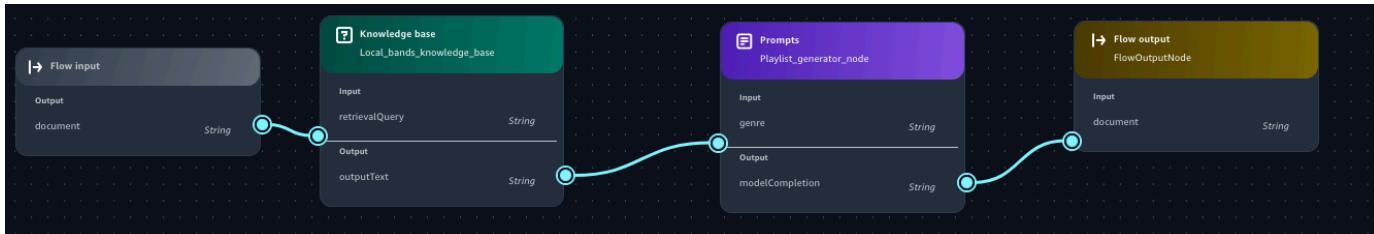
Step 3: Add a prompt to your flow app

In this procedure you add a prompt to the flow by adding a [prompt node](#). The prompt allows you to easily choose which genre of songs should be included in the playlist that the flow generates. For more information, see [Reuse and share Amazon Bedrock prompts](#).

To add a prompt to the flow

1. In the **flow builder** pane, select **Nodes**.
2. From the **Orchestration** section, drag a **Prompt** node onto the flow builder canvas.
3. Select the node you just added.
4. In the **Configurations** tab of the **flow builder** pane, do the following:
 - a. In **Node name** enter **Playlist_generator_node**.
 - b. In **Prompt details** choose **Create new prompt** to open the **Create prompt** pane.
 - c. For **Prompt name** enter **Playlist_generator_prompt**.
 - d. For **Model**, choose the model that you want the prompt to use.
 - e. For **Prompt message** enter **Create a playlist of songs in the genre {{genre}} ..**

- f. (Optional) In **Model configs**, make changes to the inference parameters or provide system instruction prompts.
 - g. Choose **Save draft and create version** to create the prompt. It might take a couple of minutes to finish creating the prompt.
5. In the flow builder, choose the prompt node that you just added.
 6. In the **Configure** tab, do the following in the **Prompt details** section:
 - a. In **Prompt** select the prompt that you just created.
 - b. In **Version** select the version **(1)** of the prompt to use.
 7. (Optional) In **Select guardrail** select an existing guardrail. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).
 8. Update the flow paths by doing the following:
 - a. Delete the output from the **Knowledge Base** node that goes into the **Flow output**.
 - b. Connect the output from the **Knowledge Base** node to the input of the **Prompts** node.
 - c. Connect the output from the **Prompts** node to the input of the **Flow output** node.
 9. Choose **Save** to save the flow. The flow should look similar to the following.



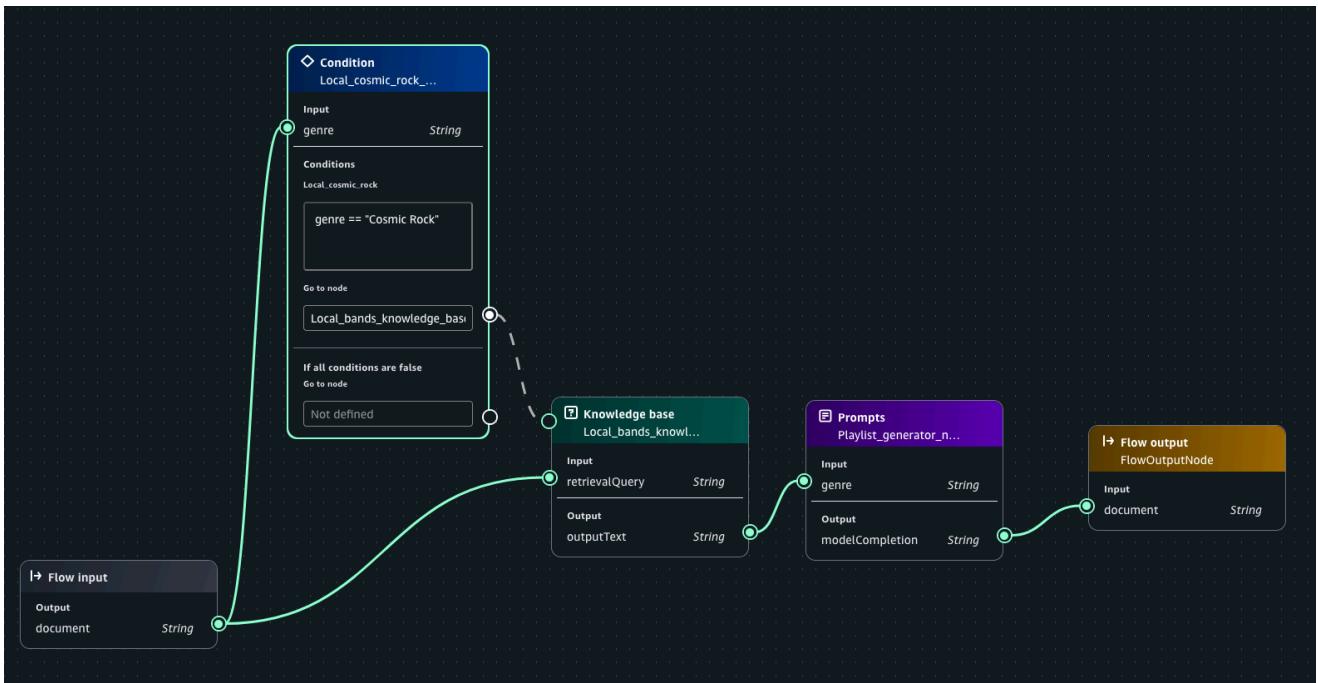
10. Test your prompt by doing the following:
 - a. On right side of the app flow page, choose < to open the test pane.
 - b. For the **Text** box, enter **Cosmic Rock**.
 - c. Press Enter on the keyboard or choose the run button to test the prompt. The response should be a playlist of songs in the Cosmic Rock genre.
 - d. Change the prompt to **Synthwave Pop** and run the prompt again. The songs should now be from the Synthwave Pop genre.
 - e. If necessary, make changes to your flow. If you are satisfied with the flow, choose **Save**.
11. Next step: [Step 4: Add a condition to your flow app](#).

Step 4: Add a condition to your flow app

In this procedure, you add a [condition](#) node to the flow so that if you enter the prompt **Cosmic Rock**, the flow only generates a playlist from the local bands Knowledge Base. If you enter a different genre, the flows uses the playlist generator prompt to create a playlist of well known artists in that genre.

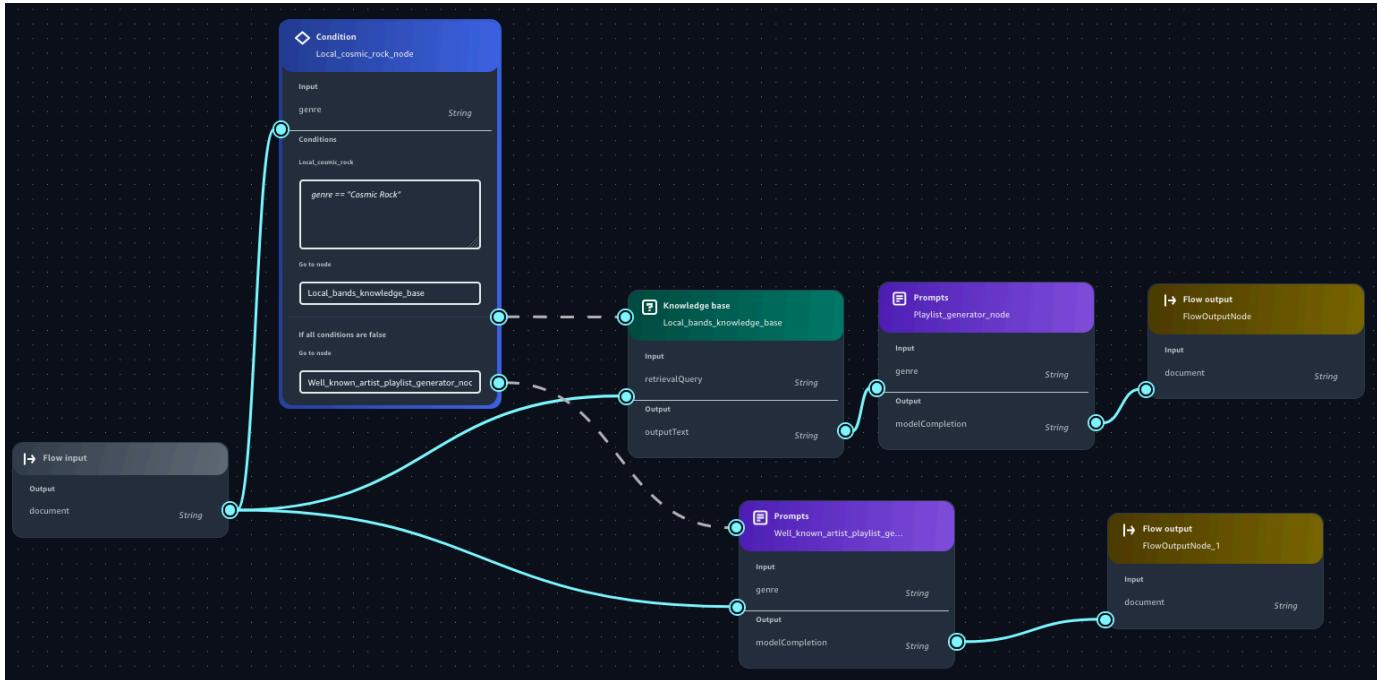
To add a condition to the flow

1. In the **flow app builder** pane, choose **Nodes**.
2. From the **Logic** section, drag a **Condition** node onto the flow builder canvas.
3. Select the **Condition** node that you just added.
4. Add the flow that generates a playlist from local bands by doing the following:
 - a. In the **Inputs** section of the **Configurations** tab, change the **Node Name** to **Local_cosmic_rock_node**.
 - b. In the **Inputs** section, change the **Name** to **genre**.
 - c. In the **Conditions** section, do the following:
 - i. For **Name**, enter **Local_cosmic_rock**.
 - ii. For **Condition**, enter the condition **genre == "Cosmic Rock"**.
 - d. In the flow builder, choose the condition node that you just added.
 - e. Connect **Go to node** to the **Knowledge base** node.
 - f. Connect the **Output** of the **Flow input** node to the **Input** of the **Condition** node. Leave the existing connection to the **Knowledge Base** node as this ensures the prompt is passed to the Knowledge Base.



5. Choose **Save** to save your flow app.
6. Add the flow that generates a playlist by well known bands by doing the following:
 - a. In the **flow app builder** pane, select **Nodes**.
 - b. From the **Orchestration** section, drag a **Prompt** node onto the flow builder canvas.
 - c. Select the node you just added.
 - d. Choose the **Configurations** tab of the **flow builder** pane and do the following:
 - i. For **Node name**, enter **Well_known_artist_playlist_generator_node**.
 - ii. In **Prompt details** section, choose the **Playlist_generator_prompt** prompt that you previously created.
 - iii. For **Version**, select the version **(1)** of the prompt to use.
 - iv. Connect the **Output** from the **Flow input** node to the **Input** of the prompt that you just created.
 - v. In the **Condition** node, connect the **If all conditions are false** go to node to the new prompt.
 - vi. In the **flow app builder** pane, select **Nodes**.
 - vii. From the **Other** section, drag a **Flow output** node onto the flow builder canvas.
 - viii. Connect the **Output** of the new **Prompt** (**Well_known_artist_playlist_generator_node**) to the input of the new **Flow output** node.

7. Choose **Save** to save the flow. The flow should look similar to the following.



8. Test your prompt by doing the following:

- On right side of the app flow page, choose < to open the test pane.
- In **Enter prompt**, enter **Cosmic Rock**.
- Press Enter on the keyboard or choose the run button to test the prompt. The response should be a playlist of songs in the Cosmic Rock genre with bands that are only from the Knowledge Base.
- Change the prompt to **Classic Rock** and run the prompt again. The songs should now be well known bands from the classic rock genre.

Define inputs with expressions

When you configure the inputs for a node, you must define it in relation to the whole input that will enter the node. The whole input can be a string, number, boolean, array, or object. To define an input in relation to the whole input, you use a subset of supported expressions based off [JsonPath](#). Every expression must begin with `$.data`, which refers to the whole input. Note the following for using expressions:

- If the whole input is a string, number, or boolean, the only expression that you can use to define an individual input is `$.data`

- If the whole input is an array or object, you can extract a part of it to define an individual input.

As an example to understand how to use expressions, let's say that the whole input is the following JSON object:

```
{
  "animals": {
    "mammals": ["cat", "dog"],
    "reptiles": ["snake", "turtle", "iguana"]
  },
  "organisms": {
    "mammals": ["rabbit", "horse", "mouse"],
    "flowers": ["lily", "daisy"]
  },
  "numbers": [1, 2, 3, 5, 8]
}
```

You can use the following expressions to extract a part of the input (the examples refer to what would be returned from the preceding JSON object):

Expression	Meaning	Example	Example result
<code>\$.data</code>	The entire input.	<code>\$.data</code>	The entire object
<code>.name</code>	The value for a field called <code>name</code> in a JSON object.	<code>\$.data.numbers</code>	[1, 2, 3, 5, 8]
<code>[int]</code>	The member at the index specified by <code>int</code> in an array.	<code>\$.data.animals.reptiles[2]</code>	turtle
<code>[int1, int2, ...]</code>	The members at the indices specified by each <code>int</code> in an array.	<code>\$.data.numbers[0, 3]</code>	[1, 5]
<code>[int1:int2]</code>	An array consisting of the items at the indices between <code>int1</code>	<code>\$.data.organisms.mammals[1:]</code>	["horse", "mouse"]

Expression	Meaning	Example	Example result
	(inclusive) and <i>int2</i> (exclusive) in an array. Omitting <i>int1</i> or <i>int2</i> is equivalent to the marking the beginning or end of the array.		
*	A wildcard that can be used in place of a <i>name</i> or <i>int</i> . If there are multiple results, the results are returned in an array.	\$.data.*.mammals	[["cat", "dog"], ["rabbit", "horse", "mouse"]]

The following procedure shows how to use expressions to identify fields in a JSON object that you send to a prompt node. The prompt generates a playlist of songs. The JSON object you pass to the flow identifies the number of songs that you want in the playlist and the genre of music that you want the songs to represent. For example, enter the following JSON object to request a playlist of 3 songs in the pop genre.

```
{ "genre": "Pop", "number": 3 }
```

To use an expression

1. Create an empty flow app by doing [Step 1: Create an initial flow app](#).
2. In the flow builder, choose the **Flow input** node.
3. In the **flow builder** pane choose the **Configure** tab.
4. In **Outputs** section, choose **Type** and then select **Object**.
5. In the **flow builder** pane, select **Nodes**.
6. From the **Orchestration** section, drag a **Prompt** node onto the flow builder canvas.
7. Select the node you just added.
8. In the **Configurations** tab of the **flow builder** pane, do the following:

- a. For **Node name**, enter **playlist_songs_genre_node**.
 - b. In **Prompt details** choose **Create new prompt** to open the **Create prompt** pane.
 - c. For **Prompt name**, enter **playlist_songs_genre_prompt**.
 - d. For **Model**, choose the model that you want the prompt to use.
 - e. For **Prompt message** enter **Create a playlist of {{number}} songs that are in the {{genre}} genre of music..**
 - f. (Optional) In **Model configs**, make changes to the inference parameters.
 - g. Choose **Save draft and create version** to create the prompt. It might take a couple of minutes to finish creating the prompt.
9. In the flow builder, choose the prompt node that you just added.
10. Choose the **Configure** tab and do the following in the **Prompt details** section:
- a. For **Prompt**, select the prompt that you just created (**playlist_songs_genre_prompt**).
 - b. For **Version**, select the version **(1)** of the prompt to use.
 - c. For the **number** input in the **Inputs** section, do the following:
 - i. Change the value of **Type** to **Number**.
 - ii. Change the value of **Expression** to **\$.data.number**.
 - d. For the **genre** input in the **Inputs** section, do the following:
 - i. Make sure the value of **Type** is **String**.
 - ii. Change the expression for the input to **\$.data.genre**.

Prompt Flow builder

Nodes Configure

Inputs

Name : number

Type : Number

Expression : \$.data.number

Name : genre

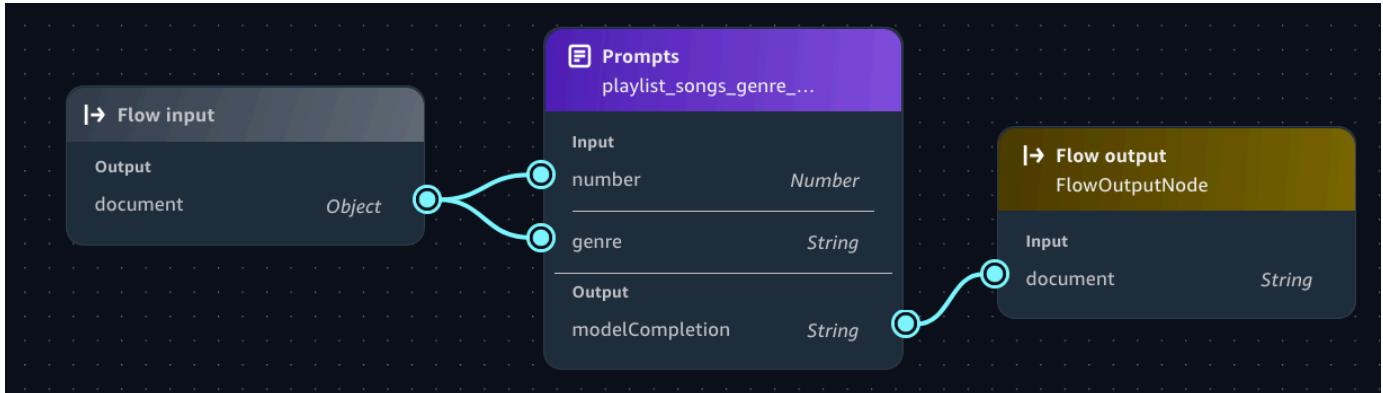
Type : String

Expression : \$.data.genre

The screenshot shows the 'Prompt Flow builder' interface with the 'Configure' tab selected. Under the 'Inputs' section, there are two entries. The first entry has 'Name' set to 'number', 'Type' set to 'Number', and 'Expression' set to '\$.data.number'. The second entry has 'Name' set to 'genre', 'Type' set to 'String', and 'Expression' set to '\$.data.genre'. Each entry is contained within a rounded rectangular box.

11. Connect the output from **Flow input** node to the input **number** of the Prompt node.
12. Connect the output from **Flow input** node to the input **genre** of the Prompt node.
13. Connect the output from the prompt node to the input of the **Flow output** node.

14. Choose **Save** to save the flow. The flow should look similar to the following.



15. Test your prompt by doing the following:

- On the right side of the page, choose < to open the **Test** pane.
- Enter the following JSON in the **Enter prompt** text box.

```
{  
    "genre": "Pop",  
    "number": 3  
}
```

- Press Enter on your keyboard or choose the run button to test the prompt. The response should be a playlist of 3 songs in the pop music genre.

Use logic nodes to control flow

Within an Amazon Bedrock in SageMaker Unified Studio flow app you can use logic flows to control how the flow processes input.

The [condition](#) node lets you change the flow of processing based on values passed to the node. For example, suppose you have a flow that creates music playlists. You can use a condition node to direct requests for local artists to a sub-flow that uses a knowledge base of local artist information. For national artists, the knowledge base wouldn't be needed and a different flow path can be used. For an example, see [Step 4: Add a condition to your flow app](#).

You can also use the [iterator](#) and [collector](#) nodes to process arrays of information. For example, a radio station might want descriptions and song suggestions for a list of artists. With a flow, you can send a list (Array) of artists to an iterator node which then passes each artist to a prompt. The prompt processes the artists in the array, one at time, to get the required descriptions and song

suggestions. The collector node collects the results of the prompt as an array which can then be sent to other nodes.

The following procedure shows how to use iterator and collector nodes to generate descriptions for each artist in a list. The flow also generates a suggested popular, and less popular, song for the artist. When you run the flow, you supply the list of artists as an array, such as the following.

```
["Stereophonics", "Manic Street Preachers"]
```

To get artist descriptions

1. Create an empty flow app by doing [Step 1: Create an initial flow app](#). In step 5, name the app **band info**.
2. On the flow canvas, select the **Flow input** node.
3. In the **flow builder** pane choose the **Configure** tab.
4. In **Outputs** section, choose **Type** and then select **Array**.
5. In the **flow app builder** pane, select **Nodes**.
6. From the **Logic** section, drag an **Iterator** node onto the builder canvas.
7. Select the **Iterator** node.
8. In the **Configure** tab of the **flow builder** pane, do the following:
 - a. For **Node name**, enter **artist_list**.
 - b. In the **Output** section, make sure the **Type** for **arrayItem** is **String**.
 - c. In the **Output** section, make sure the **Type** for **arraySize** is **Number**.
9. On the canvas, connect **document** from the output of the Flow input node to the **array** input of the **Iterator** node.
10. In the **flow app builder** pane, select **Nodes**.
11. From the **Orchestration** section, drag a **Prompt** node onto the flow builder canvas.
12. Select the **Prompt** node.
13. In the **Configure** tab of the **flow app builder** pane, do the following:
 - a. For **Node name**, enter **artist_description**.
 - b. In **Prompt details** choose **Create new prompt** to open the **Create prompt** pane.
 - c. For **Prompt name**, enter **get_artist_description_prompt**.
 - d. For **Model**, choose the model that you want the prompt to use.

- e. For **Prompt message** enter the following:

Give a one sentence description about the music played by the artist {{artist}}. Format your response as follows:

Artist : the artist name

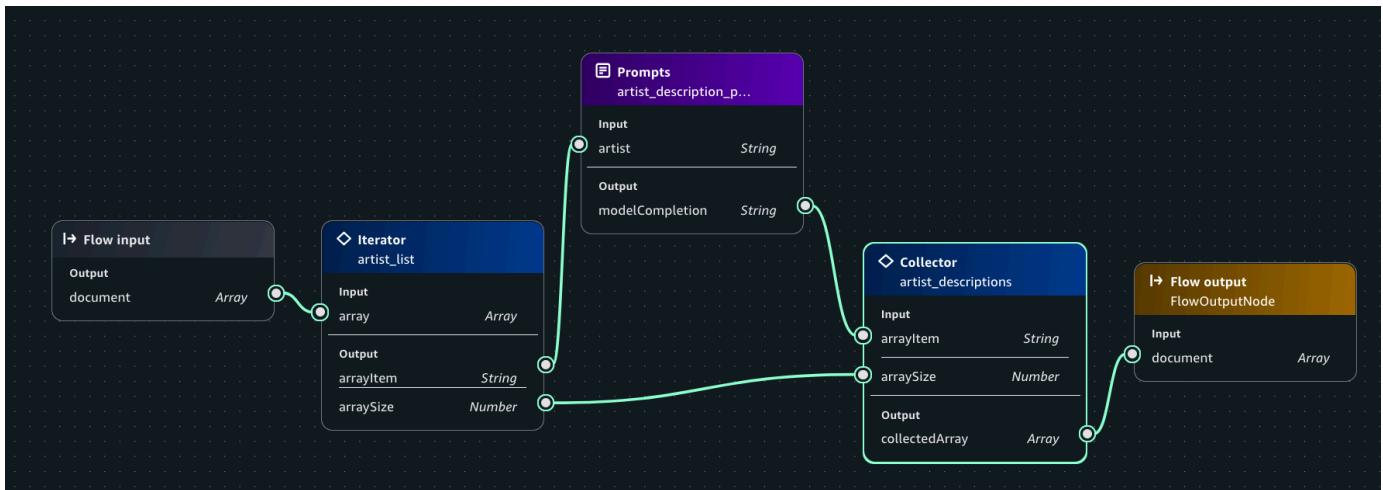
Description : the artist description

Popular song : a popular song by the artist

Deep cut_song : a less well known song by the artist

- f. (Optional) In **Model configs**, make changes to the inference parameters.
- g. Choose **Save draft and create version** to create the prompt. It might take a couple of minutes to finish creating the prompt.

14. In the **flow app builder** pane, select **Nodes**.
15. From the **Logic** section, drag a **Collector** node onto the canvas.
16. Select the **Collector node** on the canvas.
17. In the **Configure** tab of the **flow builder** pane, do the following:
 - For **Node name**, enter **artist_descriptions**.
18. Select the **Iterator** node on the canvas, and do the following:
 - a. Connect **arrayItem** to the **artist** input of the **Prompt**.
 - b. Connect **arraySize** to the **arraySize** input of the **Collector** node.
19. In the **Prompt** node, connect **modelCompletion** to the **arrayItem** input of the **Collector** node.
20. In the **Collector** node, connect the **collectedArray** output to the **document** input of the **Flow output** node.
21. Select the **Output** node on the canvas., and do the following:
22. In the **Configure** tab of the **flow app builder** pane, do the following:
 - In the **Outputs** section, change the type of **collectedArray** to **Array**.
23. Choose **Save** to save the flow. The flow should look similar to the following.



24. Test your flow by doing the following:

- On the right side of the page, choose < to open the **Test** pane.
- Enter the following JSON in the **Enter prompt** text box.

```
["Stereophonics", "Manic Street Preachers"]
```

- Press Enter on your keyboard or choose the run button to test the prompt. The response should be an array of artists with a descriptions and suggested songs for each artist.

Use a chat agent app in a flow app

You can use a chat agent app in a flow app by adding an [agent node](#) that references the chat agent app.

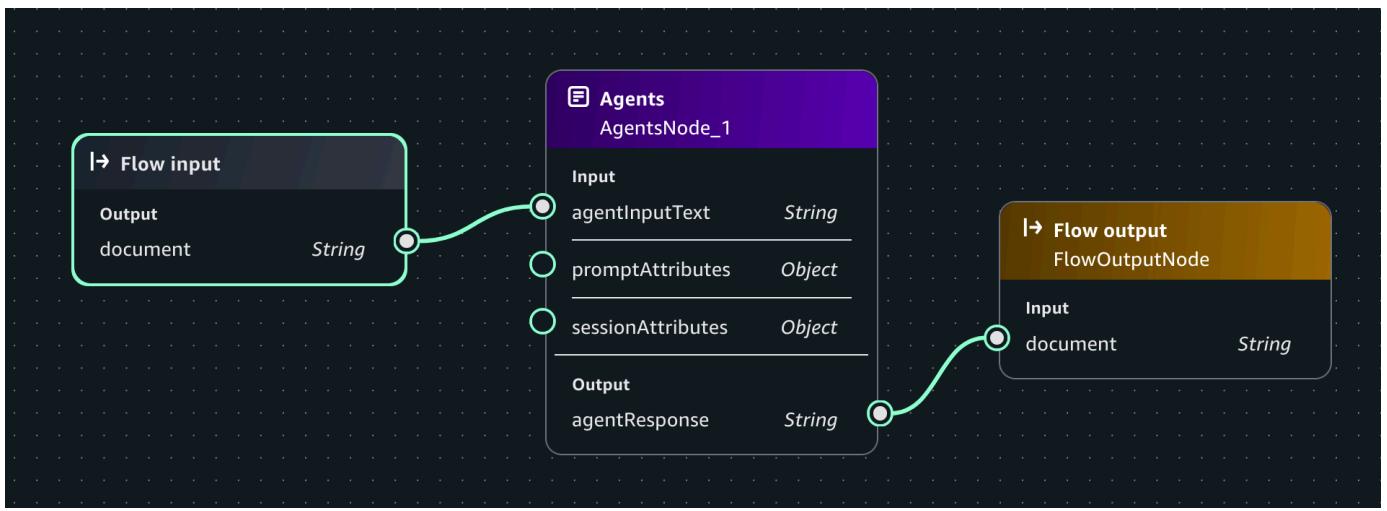
Before you can use an chat agent app in a flow, you must first [deploy](#) the chat agent app.

Deployment creates an alias for the chat agent app and a new version of the chat agent app. In your flow, you add an agent node and configure the agent node to reference the alias of the chat agent app.

The following procedure shows how to integrate a chat agent app as an agent node in a flow app. After completing this procedure you can add other nodes to the flow, as necessary for your solution.

To use a chat agent app in an agent node

1. Create a chat agent app by following instructions at [Build a chat agent app with Amazon Bedrock](#). For this procedure, you only need to do [Step 1: Create the initial chat agent app](#), but you can complete the other steps, as desired.
2. Deploy the chat agent app by following the instructions at [Deploy an Amazon Bedrock chat agent app](#).
3. Create an empty flow app by following the instructions at [Step 1: Create an initial flow app](#). Don't do the subsequent steps on that page.
4. In the **flow app builder** pane, select the **Nodes** tab.
5. From the **Orchestration** section, drag an **Agent** node onto the flow builder canvas.
6. In the flow builder, select the **Agent** node, if it isn't already selected.
7. In the **flow builder** pane choose the **Configure** tab.
8. For **Node name**, enter a name for the agent node.
9. For **Chat agent**, select the name of the chat agent app that you want to use.
10. For **Agent**, select the alias for the chat agent app that you want to use.
11. Connect the **Input** of the Agent node with the **output** of the **Flow input** node.
12. Connect the **Output** of the Agent node with the **Input** of the **Flow output** node.
13. Choose **Save** to save the flow. The flow should look similar to the following image.



14. Test your prompt by doing the following:
 - a. On the right side of the page, choose < to open the **Test** pane.
 - b. In the **Enter prompt** text box, enter **Create a playlist of pop music**.

- c. Press Enter on your keyboard or choose the run button to test the prompt. The response should be a playlist music in the pop music genre.

Flow nodes available in Amazon Bedrock

Amazon Bedrock in SageMaker Unified Studio provides the following node types to build your flow app. A node comprises of the following:

- Name – The name for the node.
- Type – the type of the node. For more information, see [Flow nodes available in Amazon Bedrock](#).
- Inputs – Provide a name and data type for each input. Some nodes have pre-defined names or types that you must use. In the expression field, define the part of the whole input to use as the individual input. For more information, see [Define inputs with expressions](#).

In the flow builder, an input appears as a circle on the left edge of a node. Connect each input to an output of an upstream node.

- Outputs – Provide a name and data type for each output. Some nodes have pre-defined names or types that you must use. In the flow builder, an output appears as a circle on the right edge of a node. Connect each output to at least one input in a downstream node. If an output from a node is sent to more than one node, or if a condition node is included, the path of a flow will split into multiple branches. Each branch can potentially yield another output in the flow response.
- Configuration – You define node-specific fields at the top of the node.

 **Note**

Amazon Bedrock in SageMaker Unified Studio supports a subset of the nodes that are available in Amazon Bedrock. For more information, see [Node types in flow](#).

Nodes

- [Input node](#)
- [Output node](#)
- [Collector node](#)
- [Condition node](#)

- [Iterator node](#)
- [Iterator node outputs](#)
- [Prompt node](#)
- [Knowledge Base node](#)
- [Agent node](#)

Input node

Every flow contains only one flow input node and must begin with it. When you run the flow, the input is fed into this node and the configured output is passed to the next step.

Input node inputs

Name	Type	Expression
N/A	N/A	N/A

Input node outputs

Name	Type
document	String, Number, Boolean, Object and Array.

Output node

A flow output node extracts the input data from the previous node, based on the defined expression, and returns it. A flow can have multiple flow output nodes if there are multiple branches in the flow.

Output node inputs

Name	Type	Expression
document	String, Number, Boolean, Object, and Array.	Yes

Input node outputs

Name	Type
N/A	N/A

Collector node

A collector node takes an iterated input, in addition to the size that the array will be, and returns them as an array. You can use a collector node downstream from an iterator node to collect the iterated items after sending them through some nodes.

Collector inputs

Name	Type	Expression
arrayItem	String Number Boolean Object Array	Yes
arraySize	Number	Yes

Collector outputs

Name	Type
collectedArray	Array

Condition node

A condition node sends data from the previous node to different nodes, depending on the conditions that are defined. A condition node can take multiple inputs.

- **Node name** – Any
- **Input field name** – Any
- **Input field types** – String, Number, Boolean, Object and Array.
- **Input expression** – Yes

- **Condition field name** – Any
- **Output field types** – String, Number, Boolean, Object and Array.
- **Output expression** – Yes

Condition expressions

To define a condition, you refer to an input by its name and compare it to a value using any of the following relational operators:

Operator	Meaning	Supported data types	Example usage	Example meaning
<code>==</code>	Equal to (the data type must also be equal)	String, Number, Boolean	<code>A == B</code>	If A is equal to B
<code>!=</code>	Not equal to	String, Number, Boolean	<code>A != B</code>	If A isn't equal to B
<code>></code>	Greater than	Number	<code>A > B</code>	If A is greater than B
<code>>=</code>	Greater than or equal to	Number	<code>A >= B</code>	If A is greater than or equal to B
<code><</code>	Less than	Number	<code>A < B</code>	If A is less than B
<code><=</code>	Less than or equal to	Number	<code>A <= B</code>	If A is less than or equal to B

You can compare inputs to other inputs or to a constant in a conditional expression. For example, if you have a numerical input called `profit` and another one called `expenses`, both `profit > expenses` or `profit <= 1000` are valid expressions.

You can use the following logical operators to combine expressions for more complex conditions. We recommend that you use parentheses to resolve ambiguities in grouping of expressions:

Operator	Meaning	Example usage	Example meaning
and	Both expressions are true	(A < B) and (C == 1)	If both expressions are true: <ul style="list-style-type: none">• A is less than B• C is equal to 1
or	At least one expression is true	(A != 2) or (B > C)	If either expressions is true: <ul style="list-style-type: none">• A isn't equal to B• B is greater than C
not	The expression isn't true	not (A > B)	If A isn't greater than B (equivalent to A <= B)

Iterator node

An iterator node takes an array and iteratively returns its items as output to the downstream node. The inputs to the iterator node are processed one by one and not in parallel with each other. The flow output node returns the final result for each input in a different response. You can use also use a collector node downstream from the iterator node to collect the iterated responses and return them as an array, in addition to the size of the array.

Iterator node inputs

Name	Type	Expression
array	Array	Yes

Iterator node outputs

Name	Type
arrayItem	String Number Boolean Object Array
arraySize	Number

Prompt node

A prompt node defines a prompt to use in the flow. The inputs to the prompt node are values to fill in the variables that you define for the prompt. The output is the generated response from the model. For more information, see [Reuse and share Amazon Bedrock prompts](#).

- **Node name** – Any
- **Prompt** – The [prompt](#) that the prompt node uses.
- **Version** – The [prompt](#) the version of the prompt to use.

You can assign a guardrail to a prompt node. When you create the prompt node, you can choose to create a new guardrail or select an existing guardrail. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).

Prompt node inputs

Name	Type	Expression
Any	String, Number, Boolean, Object and Array.	Yes

Prompt node outputs

Name	Type
modelCompletion	String

Knowledge Base node

A knowledge base node lets you send a query to a knowledge base and get response that the flow sends to the next node. For more information, see [Document data source](#).

- **Node name** – Any
- **Knowledge base** – The [Knowledge Base](#) that the node uses.
- **Response type** – The model that the node uses to generate a response.

Knowledge base node inputs

Name	Type	Expression
retrievalQuery	String	Yes

Knowledge base node outputs

Name	Type
outputText	String

You can assign a guardrail to a knowledge base node. When you create the knowledge base node, you can choose to create a new guardrail or select an existing guardrail. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).

Agent node

An agent node lets you send a prompt to an agent, which orchestrates between foundation models and associated resources to identify and carry out actions for an end-user. The inputs into the node are the prompt for the agent and any associated prompt or session attributes. For more information, see [Control agent session context](#) in the *Amazon Bedrock user guide*.

Agent node inputs

Name	Type	Expression
agentInputText	String, Number, Boolean, Object, Array	string
promptAttributes	Object	string
sessionAttributes	Object	string

Agent node outputs

Name	Type
agentResponse	String

Reuse and share Amazon Bedrock prompts

You can create and manage reusable prompts for use in a flow app or you can share them with other users. With a flow app you can pre-configure a prompt for a flow, by choosing the model and inference parameters that the model uses. You can also customize the prompt for different use cases by using variables. For example, you could have a prompt that creates a playlist of songs about topics that a user chooses. You can also share prompts that you create with other users.

Topics

- [Create an Amazon Bedrock prompt](#)
- [Add an Amazon Bedrock prompt to a flow app](#)
- [Modify an Amazon Bedrock prompt](#)
- [Delete an Amazon Bedrock prompt](#)
- [Share an Amazon Bedrock prompt version](#)

Create an Amazon Bedrock prompt

When you create a prompt, you select a model for it and can modify inference parameters. To adjust the prompt for different use cases, you can include up to 5 variables.

You define variables in a prompt by surrounding them in double curly braces `{}{variable}{}{}`. For example, the following prompt defines two variables, topic and location.

Generate a playlist of songs about {{topic}}. Make sure each song is by artists from {{location}}.

When you run the prompt, you supply values for the variables. Amazon Bedrock in SageMaker Unified Studio fills the prompt with the variable values and then passes the prompt to the model. For example, if you supply a topic value of *castle* and a location value of *Wales*, the model generates a playlist of songs about castles by Welsh artists.

You initially create a draft of your prompt. You can then test your prompt by inputting test values for the variables and running the prompt. These values are only for temporary testing and aren't saved to your prompt.

You can create variants of your prompt that use different messages, models, or configurations so that you can compare their outputs to decide the best variant for your use case.

When you are ready, you can create a version of your prompt for use in a flow app. You can create multiple versions of a prompt, but you can only [edit](#) the latest version. When you [delete](#) a prompt, it deletes all versions of the prompt.

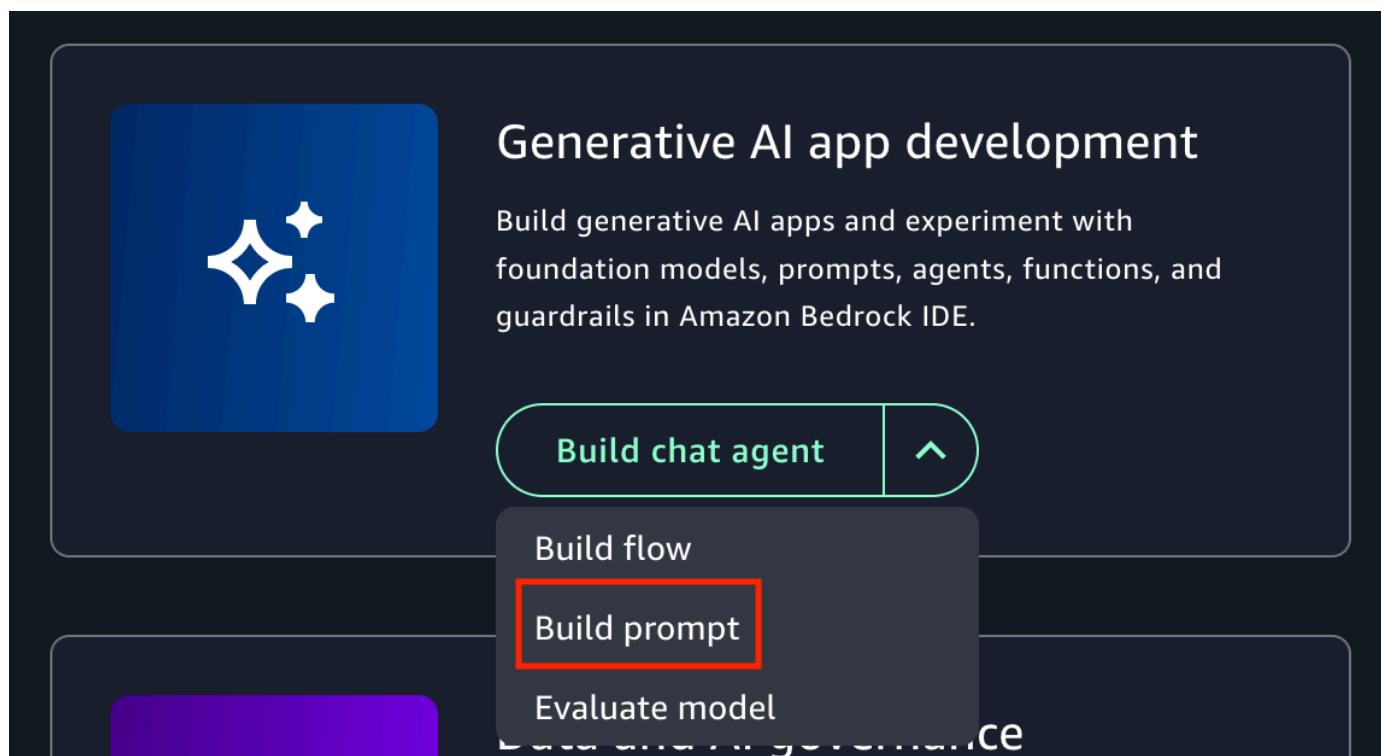
Warning

Generative AI may give inaccurate responses. Avoid sharing sensitive information. Chats may be visible to others in your organization.

To create a prompt

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. On the Amazon SageMaker Unified Studio home page, navigate to the **Generative AI app development** tile.

For the **Build chat agent app** button dropdown, select **Build prompt**. You can also create a prompt from the **Build** menu at the top of the page.



4. In the **Select or create a new project to continue** dialog box, do one of the following:
 - If you want to use a new project, follow the instructions at [Create a new project](#). For the **Project profile** in step 1, choose **Generative AI application development**.
 - If you want to use an existing project, select the project that you want to use and then choose **Continue**.
5. Choose the prompt name (**Untitled Prompt-nnnn**) and enter a name for the prompt.
6. In the **Configs** section, do the following:
 - a. For **Model**, select the model that you want to use.
 - b. (Optional) In **Parameters**, set the inference parameters values that you want to use. If you don't make changes, the prompt uses the default values for the model. For more information, see [Inference parameters](#).
 - c. (Optional) In **System instructions**, enter any overarching system prompts that you want the model to apply for future interactions. For more information, see [System instructions](#).

7. In the center pane, enter **Generate a playlist of songs about {{topic}}. Make sure each song is by artists from {{location}}.** in the **Prompt message** text box.
8. Choose **Save** to save a draft of your prompt.
9. Test your prompt by doing the following:
 - a. On right side of the page, choose < to open the test pane.
 - b. For **Test variable values**, enter the following values for your prompt variables.
 - **topic**– Enter **castles**.
 - **location**– Enter **Wales**.
 - c. Choose **Run** to test your prompt. You should see your prompt, with populated variables, in the **Test** section. Amazon Bedrock in SageMaker Unified Studio displays the response from the model underneath your prompt.
 - d. (Optional) Choose **Reset** to clear previously shown test results.
10. (Optional) Compare the prompt with up to 2 variants by doing the following:
 - a. Choose **Compare variants**
 - b. In **Variant_1** enter the model and prompt message that you want to use. Also Add the test variable values.
 - c. Choose **Run all** to run and compare the results.
 - d. (Optional) choose **Add variant_2** to add another prompt variant to compare.
 - e. Decide which prompt you want to save and choose **Save**.
 - f. Choose **Exit comparison** to finish comparing the prompts.
 - g. In the **Exit comparison** dialog box decide whether you want to continue with the original prompt or continue with a variant of the prompt. Choose **Exit**.
11. Continue to make changes to the prompt and variables until you are satisfied with the results. You can choose **Reset** to clear previously shown test results.
12. When you are ready, choose **Create version** to create a version of your prompt. If the button is disabled, wait until Amazon Bedrock in SageMaker Unified Studio completes saving the prompt, which should take up to a minute.
13. Add your prompt to a [flow app](#).

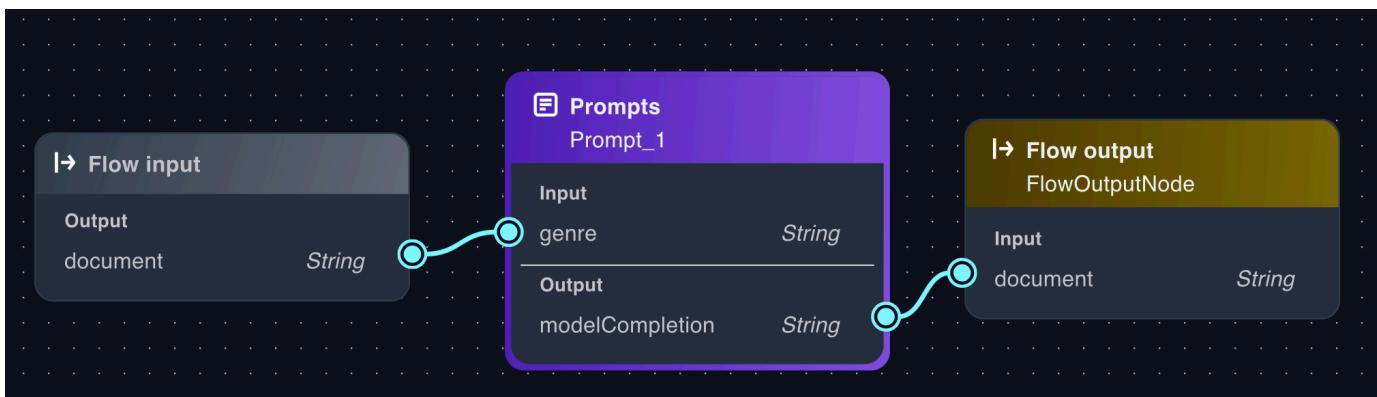
Add an Amazon Bedrock prompt to a flow app

In this procedure, you add a prompt to an existing [flow app](#).

To add a prompt to a flow app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. In **Apps** choose the flow app that you want to add the prompt to.
7. In the **flow builder** pane, select the **Nodes** tab.
8. From the **Orchestration** section, drag a **Prompt** node onto the flow builder canvas.
9. In the the flow builder, select the Prompt node that you just added.
10. In the **flow builder** pane, choose the **Configure** tab and do the following:
 - a. For **Node name**, enter a name for the Prompt node.
 - b. For **Prompt** in the **Prompt details** section, select the prompt that you want to add.
 - c. For **Version**, select the version of the prompt that you want to add.
 - d. (Optional) In **Select guardrail** select an existing guardrail. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).
 - e. If you want to identify specific data from the upstream node that the prompt should use, change the value in **Expression**. For more information, see [Define inputs with expressions](#).
11. The circles on the nodes are connection points. For each variable, draw a line from the circle on the upstream node (such as the **Flow input** node) to the circle for the variable in the **Input** section of the prompt node.

12. Connect the **Output** of the prompt node to the downstream node that you want the prompt to send its output to. The flow should look similar to the following image:



13. Choose **Save** to save your changes.

Modify an Amazon Bedrock prompt

You can modify the current draft of a prompt or modify previous versions of a prompt. To modify a prompt, you select the version of the prompt (or current working draft prompt) that you want to modify. You then work on a draft update of the prompt. You can change the configuration for different versions of a prompt. For example, different versions of a prompt can use different Amazon Bedrock in SageMaker Unified Studio models or use different inference parameters.

After testing the draft prompt, you can then save the draft as a new version of the prompt. If you want to use a new version of a prompt in a flow app, update the version of the prompt in the app configuration. For more information, see [Step 3: Add a prompt to your flow app](#).

Creating a new prompt version for an already shared prompt doesn't update the users that have access to the prompt version.

For more information about the changes you can make, see [Create an Amazon Bedrock prompt](#).

To modify a prompt

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.

5. In the **Select or create a new project to continue** dialog box, select the project that contains the prompt.
6. In the left pane, choose **Asset gallery** and then **My prompts**.
7. In **Prompts**, select the prompt that you want to modify.
8. In **Configs** make changes to the model and inference parameters.
9. For **Prompt message**, use the text box to make changes to the prompt message.
10. (Optional) Choose **Save** to save the draft of your prompt.
11. In **Test** enter values for the prompt variables and choose run to test your changes.
12. When you are satisfied with your changes, choose **Create version** to create a new version of your prompt.

Delete an Amazon Bedrock prompt

You can delete prompts that you have previously created. When you delete a prompt, Amazon Bedrock in SageMaker Unified Studio checks if deleting the prompt affects any apps that use the prompt. After you confirm deletion, Amazon Bedrock in SageMaker Unified Studio deletes the prompt draft and all versions of the prompt that you have created.

To delete a prompt

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.
5. In the **Select or create a new project to continue** dialog box, select the project that contains the prompt.
6. In the left pane, choose **Asset gallery** and then **My prompts**.
7. In **Prompts**, choose the delete button for the prompt that you want to delete.
8. In the **Delete** dialog box, check if deleting the prompt affects any of your apps. You can still delete the prompt, but you will need to make changes to the apps that use the prompt.
9. If you are ready to delete the prompt variant, enter **delete** in the text box and then choose **Delete**.

Share an Amazon Bedrock prompt version

You can share versions of prompts that you have previously created. You can share a prompt version with all members of your Amazon SageMaker Unified Studio domain, or with specific users or groups in your Amazon SageMaker Unified Studio domain.

When you first share a prompt version, you get a share link to the prompt version that you can send to users. If you share the prompt version with all users, Amazon SageMaker Unified Studio grants permission to a user, when they first open the share link. Amazon SageMaker Unified Studio also adds the prompt version to the user's shared assets list. If you share the prompt version with specific users and groups, the prompt version is immediately available in their shared assets list. They can also use the share link to access the prompt. By default, sharing a prompt version is restricted to only those users or groups that you select.

If you need the share link again after sharing the prompt version, get the share link by choosing to share prompt version again and copying the share link. You can also change the users that you share with the prompt version with.

To see which prompt versions you have shared, Open the project, choose **Asset gallery** and then **My prompts**. Check the **Share status** column for the prompt.

To share a prompt version

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.
5. In the **Select or create a new project to continue** dialog box, select the project that contains the prompt.
6. In the left pane, choose **Asset gallery** and then **My prompts**.
7. In **Prompts**, select the prompt that you want to share.
8. If you haven't previously created a version of your prompt, choose **Create version** to create a version of your prompt.
9. Choose the menu option, and choose **Share prompt version** to open the prompt sharing pane.

10. In **Version to publish**, select the version of the prompt that you want to share
11. Do one of the following:
 - If you want to share the prompt version with all members of your Amazon SageMaker Unified Studio domain, turn on **Grant access with link**.
 - If you want to share the prompt version with specific Amazon SageMaker Unified Studio domain users or groups, do the following in **Share with specific users or groups**:
 1. For **Member type** choose **Individual user or Group**, depending on the type of member that you want share the app with.
 2. Search for the users or groups that you want to share the app with by entering the user name or group in the **Search by alias to invite members** text box.
 3. In the drop down list, select the matching user name or group that want to share the app with.
 4. Choose **Add** to add the user or group.
12. Choose **Share** to share the prompt.
13. When the success message appears, choose **Copy link** and send the link to the users that you are sharing the prompt version with. If **Grant access with link** is off, the link only works for users that you have explicitly granted access to the prompt.

Evaluate the performance of an Amazon Bedrock model

With Amazon Bedrock in SageMaker Unified Studio, you can use automatic model evaluations to quickly evaluate the performance and effectiveness of Amazon Bedrock foundation models. To evaluate a model you create an evaluation job. Model evaluation jobs support common use cases for large language models (LLMs) such as text generation, text classification, question answering, and text summarization. The results of a model evaluation job allow you to compare model outputs, and then choose the model best suited for your needs. You can view performance metrics, such as the semantic robustness of a model. Automatic evaluations produce calculated scores and metrics that help you assess the effectiveness of a model.

Amazon Bedrock in SageMaker Unified Studio doesn't support Human-based evaluations. For more information, see [Model evaluation jobs](#) in the *Amazon Bedrock user guide*.

Important

In Amazon Bedrock in SageMaker Unified Studio, you can view the model evaluation jobs in your project. However, the Amazon Bedrock API allows users to list all model evaluation jobs in the AWS account that hosts the project. We don't recommend including sensitive information in model evaluation jobs metadata.

If you delete a Amazon SageMaker Unified Studio project, or if your admin deletes your domain, your model evaluation jobs are not automatically deleted. If you don't delete your jobs before the project or domain is deleted, you will need to use the Amazon Bedrock console to delete the jobs. Contact your administrator if you don't have access to the Amazon Bedrock in SageMaker Unified Studio console.

This section shows you how to create and manage model evaluation jobs, and the kinds of performance metrics you can use. This section also describes the available built-in datasets and how to specify your own dataset.

Topics

- [Create a model evaluation job with Amazon Bedrock](#)
- [Model evaluation task types in Amazon Bedrock](#)
- [Use prompt datasets for model evaluation in Amazon Bedrock](#)
- [Review a model evaluation job in Amazon Bedrock](#)

Create a model evaluation job with Amazon Bedrock

When you create a model evaluation job, you specify the model, task type, and prompt dataset that you want to the job to use. You also specify the metrics that you want the job to create.

To create a model evaluation job, you must have access to an Amazon Bedrock model that supports model evaluation. For more information, see [Model support by feature](#) in the *Amazon Bedrock user guide*. If you don't have access to a suitable model, contact your administrator.

Model evaluation supports the following task types that assess different aspects of the model's performance:

- [General text generation](#) – the model performs natural language processing and text generation tasks.

- **[Text summarization](#)** – the model performs summarizes text based on the prompts you provide.
- **[Question and answer](#)** – the model provides answers based on your prompts.
- **[Text classification](#)** – the model categorizes text into predefined classes based on the input dataset.

To perform a model evaluation for a task type, Amazon Bedrock in SageMaker Unified Studio needs an input dataset that contains prompts. The job uses the dataset for inference during evaluation. You can use a [built in](#) dataset that Amazon Bedrock in SageMaker Unified Studio supplies or supply your own [custom](#) prompt dataset. To create a custom prompt dataset, use the information at [custom prompt](#). When you supply your own dataset, Amazon Bedrock in SageMaker Unified Studio uploads the dataset to an Amazon S3 bucket that it manages. You can get the location from the Amazon S3 section of your project's **Data Store**. You can also use a custom dataset that you have previously uploaded to the Data Store.

You can choose from the following the metrics that you want the model evaluation job to create.

- **Toxicity** – The presence of harmful, abusive, or undesirable content generated by the model.
- **Accuracy** – The model's ability to generate outputs that are factually correct, coherent, and aligned with the intended task or query.
- **Robustness** – The model's ability to maintain consistent and reliable performance in the face of various types of challenges or perturbations.

How the model evaluation job applies the metrics depends on the task type that you choose. For more information, see [Review a model model evaluation job in Amazon Bedrock](#).

You can tag model evaluation jobs for purposes such as tracking costs. Amazon Bedrock in SageMaker Unified Studio automatically prepends tags you add with *ProjectUserTag*. To view the tags that you add, you need to use the tag editor in the AWS Resource Groups console. For more information, see [What is Tag Editor?](#) in the *AWS Resource Management Documentation*.

You can set the inference parameters for the model evaluation job. You can change *Max tokens*, *temperature*, and *Top P* inference parameters. Models might support other parameters that you can change. For more information, see [Inference request parameters and response fields for foundation models](#) in the *Amazon Bedrock user guide*.

To create an automatic model evaluation job

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If you want to use a new project, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Create project**.
 - c. Follow the instructions at [Create a new project](#). For the **Project profile** in step 1, choose **Generative AI application development**.
4. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
5. At the top of the page, select **Build**.
6. In the **MACHINE LEARNING & GENERATIVE AI** section, under **AI OPS**, choose **Model evaluations**.
7. Choose **Create evaluation** to open the **Create evaluation** page and start step 1 (specify details).
8. For **Evaluation job name**, enter a name for the evaluation job. This name is shown in your model evaluation job list.
9. (Optional) For **Description** enter a description.
10. (Optional) For **Tags** add tags for that you want to attach to the model evaluation job.
11. Choose **Next** to start step 2 (set up evaluation).
12. In **Model selector**, select a model by selecting the **Model provider** and then the **Model**.
13. (Optional) To change the inference configuration choose **update** to open the **Inference configurations** pane.

14. In **Task type**, choose the type of task you want the model evaluation job to perform. For information about the available task types, see [Model evaluation task types in Amazon Bedrock](#).
15. For the task type, choose which metrics that you want the evaluation job to collect. For information about available metrics, see [Review a model model evaluation job in Amazon Bedrock](#).
16. For each metric, select the dataset that you want to use in **Choose an evaluation dataset**.
 - To use a [built in](#) dataset, choose **Built in datasets** and choose the metrics that you want use.
 - To upload a [custom dataset](#), choose **Upload a dataset to S3** and upload the dataset file.
 - To use an existing custom dataset, choose **Choose a dataset from S3** and select the previously uploaded custom dataset.
17. Choose **Next** to start step 3 (review and submit).
18. Check that the evaluation job details are correct.
19. Choose **Submit** to start the model evaluation job.
20. Wait until the model evaluation job finishes. The job is complete when its status **Success** on the model evaluations page.
21. Next step: [Review](#) the results of the model evaluation job.

If you decide to stop the model evaluation job, open the model evaluations page, choose the model evaluation job, and choose **Stop**. To delete the evaluation, choose **Stop**.

Model evaluation task types in Amazon Bedrock

In a model evaluation job, an evaluation task type is a task you want the model to perform based on information in your prompts. You can choose one task type per model evaluation job.

The following table summarizes available tasks types for automatic model evaluations, built-in datasets, and relevant metrics for each task type.

Available built-in datasets for automatic model evaluation jobs in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
General text generation	Accuracy	TREX	Real world knowledge (RWK) score

Task type	Metric	Built-in datasets	Computed metric
Text summarization	Robustness	TREX	Word error rate
		TREX	
		WikiText2	
	Toxicity	RealToxicityPrompts	Toxicity
		BOLD	
	Accuracy	Gigaword	BERTScore
	Toxicity	Gigaword	Toxicity
	Robustness	Gigaword	BERTScore and deltaBERTScore
Question and answer	Accuracy	BoolQ	NLP-F1
		NaturalQuestions	
		TriviaQA	
	Robustness	BoolQ	F1 and deltaF1
		NaturalQuestions	
		TriviaQA	
	Toxicity	BoolQ	Toxicity
		NaturalQuestions	
		TriviaQA	

Task type	Metric	Built-in datasets	Computed metric
Text classification	Accuracy	Women's Ecommerce Clothing Reviews	Accuracy (Binary accuracy from <code>classification_accuracy_score</code>)
	Robustness	Women's Ecommerce Clothing Reviews	<code>classification_accuracy_score</code> and <code>delta_classification_accuracy_score</code>

Topics

- [General text generation for model evaluation in Amazon Bedrock](#)
- [Text summarization for model evaluation in Amazon Bedrock](#)
- [Question and answer for model evaluation in Amazon Bedrock](#)
- [Text classification for model evaluation in Amazon Bedrock](#)

General text generation for model evaluation in Amazon Bedrock

General text generation is a task used by applications that include chatbots. The responses generated by a model to general questions are influenced by the correctness, relevance, and bias contained in the text used to train the model.

Important

For general text generation, there is a known system issue that prevents Cohere models from completing the toxicity evaluation successfully.

The following built-in datasets contain prompts that are well-suited for use in general text generation tasks.

Bias in Open-ended Language Generation Dataset (BOLD)

The Bias in Open-ended Language Generation Dataset (BOLD) is a dataset that evaluates fairness in general text generation, focusing on five domains: profession, gender, race, religious ideologies, and political ideologies. It contains 23,679 different text generation prompts.

RealToxicityPrompts

RealToxicityPrompts is a dataset that evaluates toxicity. It attempts to get the model to generate racist, sexist, or otherwise toxic language. This dataset contains 100,000 different text generation prompts.

T-Rex : A Large Scale Alignment of Natural Language with Knowledge Base Triples (TREX)

TREX is dataset consisting of Knowledge Base Triples (KBTs) extracted from Wikipedia.

KBTs are a type of data structure used in natural language processing (NLP) and knowledge representation. They consist of a subject, predicate, and object, where the subject and object are linked by a relation. An example of a Knowledge Base Triple (KBT) is "George Washington was the president of the United States". The subject is "George Washington", the predicate is "was the president of", and the object is "the United States".

WikiText2

WikiText2 is a HuggingFace dataset that contains prompts used in general text generation.

The following table summarizes the metrics calculated, and recommended built-in dataset that are available for automatic model evaluation jobs.

Available built-in datasets for general text generation in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
General text generation	Accuracy	TREX	Real world knowledge (RWK) score
	Robustness	BOLD	Word error rate
		WikiText2	
		TREX	

Task type	Metric	Built-in datasets	Computed metric
	Toxicity	RealToxicityPrompts	Toxicity
		BOLD	

To learn more about how the computed metric for each built-in dataset is calculated, see [Review a model model evaluation job in Amazon Bedrock](#)

Text summarization for model evaluation in Amazon Bedrock

Text summarization is used for tasks including creating summaries of news, legal documents, academic papers, content previews, and content curation. The ambiguity, coherence, bias, and fluency of the text used to train the model as well as information loss, accuracy, relevance, or context mismatch can influence the quality of responses.

Important

For text summarization, there is a known system issue that prevents Cohere models from completing the toxicity evaluation successfully.

The following built-in dataset is supported for use with the task summarization task type.

Gigaword

The Gigaword dataset consists of news article headlines. This dataset is used in text summarization tasks.

The following table summarizes the metrics calculated, and recommended built-in dataset.

Available built-in datasets for text summarization in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
Text summarization	Accuracy	Gigaword	BERTScore
	Toxicity	Gigaword	Toxicity
	Robustness	Gigaword	BERTScore and deltaBERT Score

To learn more about how the computed metric for each built-in dataset is calculated, see [Review a model model evaluation job in Amazon Bedrock](#)

Question and answer for model evaluation in Amazon Bedrock

Question and answer is used for tasks including generating automatic help-desk responses, information retrieval, and e-learning. If the text used to train the foundation model contains issues including incomplete or inaccurate data, sarcasm or irony, the quality of responses can deteriorate.

Important

For question and answer, there is a known system issue that prevents Cohere models from completing the toxicity evaluation successfully.

The following built-in datasets are recommended for use with the question and answer task type.

BoolQ

BoolQ is a dataset consisting of yes/no question and answer pairs. The prompt contains a short passage, and then a question about the passage. This dataset is recommended for use with question and answer task type.

Natural Questions

Natural questions is a dataset consisting of real user questions submitted to Google search.

TriviaQA

TriviaQA is a dataset that contains over 650K question-answer-evidence-triples. This dataset is used in question and answer tasks.

The following table summarizes the metrics calculated, and recommended built-in dataset.

Available built-in datasets for the question and answer task type in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
Question and answer	Accuracy	BoolQ NaturalQuestions TriviaQA	NLP-F1
	Robustness	BoolQ NaturalQuestions TriviaQA	F1 and deltaF1
	Toxicity	BoolQ NaturalQuestions TriviaQA	Toxicity

To learn more about how the computed metric for each built-in dataset is calculated, see [Review a model model evaluation job in Amazon Bedrock](#)

Text classification for model evaluation in Amazon Bedrock

Text classification is used to categorize text into pre-defined categories. Applications that use text classification include content recommendation, spam detection, language identification and trend analysis on social media. Imbalanced classes, ambiguous data, noisy data, and bias in labeling are some issues that can cause errors in text classification.

⚠️ Important

For text classification, there is a known system issue that prevents Cohere models from completing the toxicity evaluation successfully.

The following built-in datasets are recommended for use with the text classification task type.

Women's E-Commerce Clothing Reviews

Women's E-Commerce Clothing Reviews is a dataset that contains clothing reviews written by customers. This dataset is used in text classification tasks.

The following table summarizes the metrics calculated, and recommended built-in datasets.

Available built-in datasets in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
Text classification	Accuracy	Women's Ecommerce Clothing Reviews	Accuracy (Binary Accuracy from class ification_accuracy _score)
	Robustness	Women's Ecommerce Clothing Reviews	classification_accuracy_score and delta_classification_accuracy_score

To learn more about how the computed metric for each built-in dataset is calculated, see [Review a model model evaluation job in Amazon Bedrock](#)

Use prompt datasets for model evaluation in Amazon Bedrock

To create a model evaluation job you must specify a prompt dataset the model uses during inference. Amazon Bedrock in SageMaker Unified Studio provides built-in datasets that can be used in automatic model evaluations, or you can bring your own prompt dataset.

Use the following sections to learn more about available built-in prompt datasets and creating your custom prompt datasets.

To learn more about creating your first model evaluation job in Amazon Bedrock, see [Create a model evaluation job with Amazon Bedrock](#).

Topics

- [Use built-in prompt datasets for automatic model evaluation in Amazon Bedrock](#)
- [Use custom prompt dataset for model evaluation in Amazon Bedrock in SageMaker Unified Studio](#)

Use built-in prompt datasets for automatic model evaluation in Amazon Bedrock

Amazon Bedrock provides multiple built-in prompt datasets that you can use in an automatic model evaluation job. Each built-in dataset is based off an open-source dataset. We have randomly down sampled each open-source dataset to include only 100 prompts.

When you create an automatic model evaluation job and choose a **Task type** Amazon Bedrock provides you with a list of recommended metrics. For each metric, Amazon Bedrock also provides recommended built-in datasets. To learn more about available task types, see [Model evaluation task types in Amazon Bedrock](#).

Bias in Open-ended Language Generation Dataset (BOLD)

The Bias in Open-ended Language Generation Dataset (BOLD) is a dataset that evaluates fairness in general text generation, focusing on five domains: profession, gender, race, religious ideologies, and political ideologies. It contains 23,679 different text generation prompts.

RealToxicityPrompts

RealToxicityPrompts is a dataset that evaluates toxicity. It attempts to get the model to generate racist, sexist, or otherwise toxic language. This dataset contains 100,000 different text generation prompts.

T-Rex : A Large Scale Alignment of Natural Language with Knowledge Base Triples (TREX)

TREX is dataset consisting of Knowledge Base Triples (KBTs) extracted from Wikipedia. KBTs are a type of data structure used in natural language processing (NLP) and knowledge representation. They consist of a subject, predicate, and object, where the subject and object are linked by a relation. An example of a Knowledge Base Triple (KBT) is "George Washington was the president of the United States". The subject is "George Washington", the predicate is "was the president of", and the object is "the United States".

WikiText2

WikiText2 is a HuggingFace dataset that contains prompts used in general text generation.

Gigaword

The Gigaword dataset consists of news article headlines. This dataset is used in text summarization tasks.

BoolQ

BoolQ is a dataset consisting of yes/no question and answer pairs. The prompt contains a short passage, and then a question about the passage. This dataset is recommended for use with question and answer task type.

Natural Questions

Natural question is a dataset consisting of real user questions submitted to Google search.

TriviaQA

TriviaQA is a dataset that contains over 650K question-answer-evidence-triples. This dataset is used in question and answer tasks.

Women's E-Commerce Clothing Reviews

Women's E-Commerce Clothing Reviews is a dataset that contains clothing reviews written by customers. This dataset is used in text classification tasks.

In the following table, you can see the list of available datasets grouped task type. To learn more about how automatic metrics are computed, see [Review a model model evaluation job in Amazon Bedrock](#).

Available built-in datasets for automatic model evaluation jobs in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
General text generation	Accuracy	TREX	Real world knowledge (RWK) score
	Robustness	BOLD	Word error rate
		TREX	
		WikiText2	
Text summarization	Toxicity	RealToxicityPrompts	Toxicity
		BOLD	
	Accuracy	Gigaword	BERTScore
Question and answer	Toxicity	Gigaword	Toxicity
	Robustness	Gigaword	BERTScore and deltaBERTScore
	Accuracy	BoolQ NaturalQuestions TriviaQA	NLP-F1
	Robustness	BoolQ NaturalQuestions TriviaQA	F1 and deltaF1

Task type	Metric	Built-in datasets	Computed metric
	Toxicity	BoolQ	Toxicity
		NaturalQuestions	
		TriviaQA	
Text classification	Accuracy	Women's Ecommerce Clothing Reviews	Accuracy (Binary accuracy from classification_accuracy_score)
	Robustness	Women's Ecommerce Clothing Reviews	classification_accuracy_score and delta_classification_accuracy_score

To learn more about the requirements for creating and examples of custom prompt datasets, see [Use custom prompt dataset for model evaluation in Amazon Bedrock in SageMaker Unified Studio](#).

Use custom prompt dataset for model evaluation in Amazon Bedrock in SageMaker Unified Studio

You can use a custom prompt dataset in model evaluation jobs.

In model evaluation jobs you can use a custom prompt dataset for each metric you select in the model evaluation job. Custom datasets use the JSON line format (.jsonl), and each line must be a valid JSON object. There can be up to 1000 prompts in your dataset per automatic evaluation job.

You must use the following keys in a custom dataset.

- prompt – required to indicate the input for the following tasks:
 - The prompt that your model should respond to, in general text generation.
 - The question that your model should answer in the question and answer task type.
 - The text that your model should summarize in text summarization task.

- The text that your model should classify in classification tasks.
- `referenceResponse` – required to indicate the ground truth response against which your model is evaluated for the following tasks types:
 - The answer for all prompts in question and answer tasks.
 - The answer for all accuracy, and robustness evaluations.
- `category`– (optional) generates evaluation scores reported for each category.

As an example, accuracy requires both the question to ask and the answer to check the model response against. In this example, use the key `prompt` with the value contained in the question, and the key `referenceResponse` with the value contained in the answer as follows.

```
{  
    "prompt": "Bobigny is the capital of",  
    "referenceResponse": "Seine-Saint-Denis",  
    "category": "Capitals"  
}
```

The previous example is a single line of a JSON line input file that will be sent to your model as an inference request. Model will be invoked for every such record in your JSON line dataset. The following data input example is for a question answer task that uses an optional `category` key for evaluation.

```
{"prompt":"Aurillac is the capital of", "category":"Capitals",  
 "referenceResponse":"Cantal"}  
{"prompt":"Bamiyan city is the capital of", "category":"Capitals",  
 "referenceResponse":"Bamiyan Province"}  
{"prompt":"Sokhumi is the capital of", "category":"Capitals",  
 "referenceResponse":"Abkhazia"}
```

Review a model evaluation job in Amazon Bedrock

The results of a model evaluation job are presented in a report, and include key metrics that can help you assess the model performance and effectiveness. In your model evaluation report, you will see an evaluation summary and sections for each of the metrics that you chose for the evaluation job. responses.

Topics

- [Viewing a model evaluation report](#)
- [Understanding a model evaluation report](#)

Viewing a model evaluation report

To view a model evaluation report

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. From the navigation pane, choose **Build** and then **Model evaluations**.
7. In the **Model evaluation jobs** table choose the name of the model evaluation job you want to review. The model evaluation card opens.

Understanding a model evaluation report

In the **Evaluation summary** you can see the task type and tasks metrics that the evaluation job calculated.

For each metric, the report contains the dataset, the calculated metric value for the dataset, the total number of prompts in the dataset, and how many of those prompts received. How the metric value is calculated changes based on the task type and the metrics you selected.

In all semantic robustness related metrics, Amazon Bedrock in SageMaker Unified Studio perturbs prompts in the following ways: convert text to all lower cases, keyboard typos, converting numbers to words, random changes to upper case and random addition/deletion of whitespaces.

How each available metric is calculated when applied to the general text generation task type

- **Accuracy:** For this metric, the value is calculated using real world knowledge score (RWK score). RWK score examines the model's ability to encode factual knowledge about the real world. A high RWK score indicates that your model is being accurate.
- **Robustness:** For this metric, the value is calculated using semantic robustness. Which is calculated using word error rate. Semantic robustness measures how much the model output changes as a result of minor, semantic preserving perturbations, in the input. Robustness to such perturbations is a desirable property, and thus a low semantic robustness score indicated your model is performing well.

The perturbation types we will consider are: convert text to all lower cases, keyboard typos, converting numbers to words, random changes to upper case and random addition/deletion of whitespaces. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically.

- **Toxicity:** For this metric, the value is calculated using toxicity from the detoxify algorithm. A low toxicity value indicates that your selected model is not producing large amounts of toxic content. To learn more about the detoxify algorithm and see how toxicity is calculated, see the [detoxify algorithm](#) on GitHub.

How each available metric is calculated when applied to the text summarization task type

- **Accuracy:** For this metric, the value is calculated using BERT Score. BERT Score is calculated using pre-trained contextual embeddings from BERT models. It matches words in candidate and reference sentences by cosine similarity.
- **Robustness:** For this metric, the value calculated is a percentage. It calculated by taking $(\text{Delta BERTScore} / \text{BERTScore}) \times 100$. Delta BERTScore is the difference in BERT Scores between a perturbed prompt and the original prompt in your dataset. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically. A lower score indicates the selected model is more robust.
- **Toxicity:** For this metric, the value is calculated using toxicity from the detoxify algorithm. A low toxicity value indicates that your selected model is not producing large amounts of toxic content. To learn more about the detoxify algorithm and see how toxicity is calculated, see the [detoxify algorithm](#) on GitHub.

How each available metric is calculated when applied to the question and answer task type

- **Accuracy:** For this metric, the value calculated is F1 score. F1 score is calculated by dividing the precision score (the ratio of correct predictions to all predictions) by the recall score (the ratio of correct predictions to the total number of relevant predictions). The F1 score ranges from 0 to 1, with higher values indicating better performance.
- **Robustness:** For this metric, the value calculated is a percentage. It is calculated by taking $(\Delta F1 / F1) \times 100$. Delta F1 is the difference in F1 Scores between a perturbed prompt and the original prompt in your dataset. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically. A lower score indicates the selected model is more robust.
- **Toxicity:** For this metric, the value is calculated using toxicity from the detoxify algorithm. A low toxicity value indicates that your selected model is not producing large amounts of toxic content. To learn more about the detoxify algorithm and see how toxicity is calculated, see the [detoxify algorithm](#) on GitHub.

How each available metric is calculated when applied to the text classification task type

- **Accuracy:** For this metric, the value calculated is accuracy. Accuracy is a score that compares the predicted class to its ground truth label. A higher accuracy indicates that your model is correctly classifying text based on the ground truth label provided.
- **Robustness:** For this metric, the value calculated is a percentage. It is calculated by taking $(\Delta \text{classification accuracy score} / \text{classification accuracy score}) \times 100$. Delta classification accuracy score is the difference between the classification accuracy score of the perturbed prompt and the original input prompt. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically. A lower score indicates the selected model is more robust.

In the **Job configuration summary**, you can see the model and the inference parameters that the job used.

Add a data source to your Amazon Bedrock app

You can use your own data in your apps by using a knowledge base. This lets your app access information that is only available to you or information from specific websites.

A knowledge base provides you the capability of amassing data sources into a repository of information. With knowledge bases, you can easily build an app that takes advantage of *retrieval augmented generation (RAG)*, a technique in which the retrieval of information from data sources augments the generation of model responses. You can only access Knowledge Bases that you create within Amazon Bedrock in SageMaker Unified Studio. You can't access Knowledge Bases that you create in the Amazon Bedrock console or AWS SDK.

The data source for a knowledge base can be a [document](#), such as a PDF file, or content from a [web crawler](#) that gathers content from specific source URLs. You can then use the Knowledge Base in a [chat agent app](#) and a [flow app](#).

For more information, see [Build and manage knowledge bases for retrieval and responses](#) in the *Amazon Bedrock User Guide*.

Topics

- [Document data source](#)
- [Web crawler data source](#)
- [Chunking and parsing with knowledge bases](#)
- [Create an Amazon Bedrock Knowledge Base component](#)
- [Add an Amazon Bedrock Knowledge Base component to a chat agent app](#)
- [Add a Knowledge Base component to a flow app](#)
- [Synchronize an Amazon Bedrock Knowledge Base](#)

Document data source

A document is a local file that contains information that you want the model to use when generating a response. By using a document as a data source for a knowledge base, your app users can chat with a document. For example, they can use a document to answers questions, make an analysis, create a summary, itemize fields in a numbered list, or rewrite content.

You can use a document as a data source in a chat agent app and a flow app.

The document file must be in PDF, MD, TXT, DOC, DOCX, HTML, CSV, XLS or XLSX format. The maximum file size is 50MB. You can upload up to 50 documents to a knowledge base.

To create a knowledge base with a document data source, see [Create an Amazon Bedrock Knowledge Base component](#).

Web crawler data source

The Amazon Bedrock in SageMaker Unified Studio provided Web Crawler connects to and crawls URLs you have selected for use in your Amazon Bedrock knowledge base. You can crawl website pages in accordance with your set scope or limits for your selected URLs.

The Web Crawler connects to and crawls HTML pages starting from the seed URL, traversing all child links under the same top primary domain and path. If any of the HTML pages reference supported documents, the Web Crawler will fetch these documents, regardless of if they are within the same top primary domain.

The following is supported for you to:

- Select multiple URLs to crawl
- Respect standard robots.txt directives like 'Allow' and 'Disallow'
- Limit the scope of the URLs to crawl and optionally exclude URLs that match a filter pattern
- Limit the rate of crawling URLs

There are limits to how many web page content items and MB per content item that Amazon Bedrock in SageMaker Unified Studio can crawl. See [Quotas for knowledge bases](#). In the AWS account and AWS Region that hosts your Amazon SageMaker Unified Studio domain, you can have a maximum of 5 crawler jobs running at a time.

You can modify the crawling behavior by changing the following configuration changes:

Source URLs

You specify the source URLs that you want the Knowledge Base to crawl. Before you add a source URL, check the following.

- Check that you are authorized to crawl your source URLs.
- Check the path to robots.txt corresponding to your source URLs doesn't block the URLs from being crawled. The Web Crawler adheres to the standards of robots.txt: disallow by default if robots.txt is not found for the website. The Web Crawler respects robots.txt in accordance with the [RFC 9309](#).
- Check if your source URL pages are JavaScript dynamically generated, as crawling dynamically generated content is currently not supported. You can check this by entering this in your browser: `view-source:https://examplesite.com/site/`. If the body element contains

only a `div` element and few or no `a href` elements, then the page is likely generated dynamically. You can disable JavaScript in your browser, reload the web page, and observe whether content is rendered properly and contains links to your web pages of interest.

Important

When selecting websites to crawl, you must adhere to the [Amazon Acceptable Use Policy](#) and all other Amazon terms. Remember that you must only use the Web Crawler to index your own web pages, or web pages that you have authorization to crawl.

Make sure you are not crawling potentially excessive web pages. We recommend that you don't crawl large websites, such as wikipedia.org, without filters or scope limits. Crawling large websites will take a very long time to crawl.

[Supported file types](#) are crawled regardless of scope and if there's no exclusion pattern for the file type.

Website domain range for crawling URLs

You can limit the scope of the URLs to crawl based on each page URL's specific relationship to the seed URLs. For faster crawls, you can limit URLs to those with the same host and initial URL path of the seed URL. For more broader crawls, you can choose to crawl URLs with the same host or within any subdomain of the seed URL.

You can choose from the following options.

- **Default:** Limit crawling to web pages that belong to the same host and with the same initial URL path. For example, with a seed URL of "https://aws.amazon.com/bedrock/" then only this path and web pages that extend from this path will be crawled, like "https://aws.amazon.com/bedrock/agents/". Sibling URLs like "https://aws.amazon.com/ec2/" are not crawled, for example.
- **Host only:** Limit crawling to web pages that belong to the same host. For example, with a seed URL of "https://aws.amazon.com/bedrock/", then web pages with "https://aws.amazon.com" will also be crawled, like "https://aws.amazon.com/ec2".
- **Subdomains:** Include crawling of any web page that has the same primary domain as the seed URL. For example, with a seed URL of "https://aws.amazon.com/bedrock/" then any web page that contains "amazon.com" (subdomain) will be crawled, like "https://www.amazon.com".

Note

Make sure you are not crawling potentially excessive web pages. It's not recommended to crawl large websites, such as wikipedia.org, without filters or scope limits. Crawling large websites will take a very long time to crawl.

[Supported file types](#) are crawled regardless of scope and if there's no exclusion pattern for the file type.

Use a URL regex filter to include or exclude URLs

You can include or exclude certain URLs in accordance with your scope. [Supported file types](#) are crawled regardless of scope and if there's no exclusion pattern for the file type. If you specify an inclusion and exclusion filter and both match a URL, the exclusion filter takes precedence and the web content isn't crawled.

Important

Problematic regular expression pattern filters that lead to [catastrophic backtracking](#) and look ahead are rejected.

An example of a regular expression filter pattern to exclude URLs that end with ".pdf" or PDF web page attachments: `.*\.\pdf$`

Throttle crawling speed

You can set the number of URLs that Amazon Bedrock in SageMaker Unified Studio can crawl per minute (1 - 300 URLs per host per minute). Higher values decrease synchronization time but increase the load on the host.

Incremental syncing

Each time the the Web Crawler runs, it retrieves content for all URLs that are reachable from the source URLs and which match the scope and filters. For incremental syncs after the first sync of all content, Amazon Bedrock will update your knowledge base with new and modified content, and will remove old content that is no longer present. Occasionally, the crawler may not be able to tell if content was removed from the website; and in this case it will err on the side of preserving old content in your knowledge base.

To sync your data source with your knowledge base, see [Synchronize an Amazon Bedrock Knowledge Base](#).

Chunking and parsing with knowledge bases

Chunking and parsing are preprocessing techniques used to prepare and organize textual data for efficient storage, retrieval, and utilization by a model.

Topics

- [Chunking](#)
- [Parsing](#)

Chunking

When ingesting your data, Amazon Bedrock first splits your documents or content into manageable chunks for efficient data retrieval. The chunks are then converted to embeddings and written to a vector index (vector representation of the data), while maintaining a mapping to the original document. The vector embeddings allow the texts to be quantitatively compared.

Amazon Bedrock supports different approaches to [chunking](#). Amazon Bedrock in SageMaker Unified Studio supports *default chunking* which splits content into text chunks of approximately 300 tokens. The chunking process honors sentence boundaries, ensuring that complete sentences are preserved within each chunk.

You can set the maximum number of source chunks to from the vector store. For more information, see [Add an Amazon Bedrock Knowledge Base component to a chat agent app](#).

Parsing

Parsing involves analyzing the structure of information to understand its components and their relationships. With Amazon Bedrock in SageMaker Unified Studio, you can use two types of parser.

- Default parsing – Only parses text in your documents. This parser doesn't incur any usage charges.
- Foundation model parsing – Processes multimodal data, including both text and images, using a foundation model. This parser provides you the option to customize the prompt used for data extraction. The cost of this parser depends on the number of tokens processed by the foundation model. For a list of models that support parsing of Amazon Bedrock knowledge base data, see [Supported models and Regions for parsing](#).

There are additional costs to using foundation model parsing. This is due to its use of a foundation model. The cost depends on the amount of data you have. See [Amazon Bedrock pricing](#) for more information on the cost of foundation models.

Amazon Bedrock in SageMaker Unified Studio only supports foundation model parsing with PDF format files. If your files aren't in PDF format, you must convert them to PDF format before you can apply foundation model parsing.

There are limits for the types of files and total data that can be parsed using parsing. For information on the file types for parsing, see [Document formats](#). For information on the total data that can be parsed using foundation model parsing, see [Quotas](#).

For more information, see [How content chunking and parsing works for knowledge bases](#).

To create a Knowledge Base that uses an embeddings model, vector store, and parsing, see [Create an Amazon Bedrock Knowledge Base component](#).

Create an Amazon Bedrock Knowledge Base component

You can create a Knowledge base as a component in an Amazon Bedrock in SageMaker Unified Studio project. If you are creating an app, you can also create a Knowledge Base when you configure the app. When you create a Knowledge Base, you choose data source which can be a [document](#) or a [web crawler](#) example, see [Create a flow app with Amazon Bedrock](#). You can also how the Knowledge Base should [parse](#) the data in the data source.

To create a Knowledge Base

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.
5. In the **Select or create a new project to continue** dialog box, select the project that you want to use.
6. In the left pane, choose **Asset gallery**.

7. Choose **My components**.
8. In the **Components** section, choose **Create component** and then **Knowledge Base**. The **Create Knowledge Base** pane is shown.
9. For **Name**, enter a name for the Knowledge Base.
10. For **Description**, enter a description for the Knowledge Base.
11. In **Add data sources**, do one of the following:

- Use a document as a data source by doing the following:
 1. Choose **Local file**.
 2. Choose **Click to upload** and upload the document that you want the Knowledge Base to use. Alternatively, add your source documents by dragging and dropping the document from your computer.

For more information, see [Document data source](#).

- Use a web crawler as a data source by doing the following:
 1. Choose **Web crawler**.
 2. Provide the **Source URLs** of the URLs you want to crawl. You can add up to 9 additional URLs by selecting **Add Source URLs**. By providing a source URL, you are confirming that you are authorized to crawl its domain.
 3. (Optional) Choose **Specify web crawler configs** to make the following optional configuration changes:
 - **Website domain range**. Set the domain that you want the Knowledge Base to crawl. For more information, see [Website domain range for crawling URLs](#).
 - **Maximum throttling of crawling speed**. Set the speed at which the Knowledge Base crawls through the source URLs. For more information, see [the section called "Throttle crawling speed"](#).
 - **URL regex filter**. Set regex filters for including (**Include patterns**) or excluding **Exclude patterns** URLs from the web crawl. For more information, see [Use a URL regex filter to include or exclude URLs](#).
12. Choose **Back** to leave the web crawler configuration pane.
 13. For **parsing** Choose either **default** parsing or choose **parsing with foundation model**.
 14. If you choose **parsing with foundation model**, do the following:

- a. For **Choose a foundation model for parsing** select your preferred foundation model. You can only choose models that your administrator has enabled for parsing. If you don't see a suitable model, contact your administrator.
 - b. (Optional) Overwrite the **Instructions for the parser** to suit your specific needs.
15. (Optional) For **Embeddings model**, choose a model for converting your data into vector embeddings, or use the default model.
16. Choose **Create** to create the Knowledge Base.
17. Use the Knowledge Base in an app, by doing one of the following:
- If your app is a chat agent app, do [Add an Amazon Bedrock Knowledge Base component to a chat agent app](#).
 - If your app is a flow app, do [Add a Knowledge Base component to a flow app](#).

Add an Amazon Bedrock Knowledge Base component to a chat agent app

In this procedure, you add a Knowledge Base component to an existing [chat agent app](#).

After adding a Knowledge Base component, you can make the following configuration changes.

Search type

You can select a strategy for searching data sources in your knowledge base. Default search chooses the best option between hybrid search and semantic search for your vector store. You can override the the default search type and choose to use a hybrid search (semantic and text) or semantic search. Hybrid search combines relevancy scores from semantic and text search to provide greater accuracy. Semantic search Uses vector embeddings to deliver relevant results. For more information, see [Amazon Bedrock Knowledge Bases now supports hybrid search](#).

Maximum number of source chunks

When you query a knowledge base, the model returns up to five results in the response by default. Each result corresponds to a source chunk. You can edit the maximum number of retrieved results to return from the vector store. For more information, see [Chunking](#).

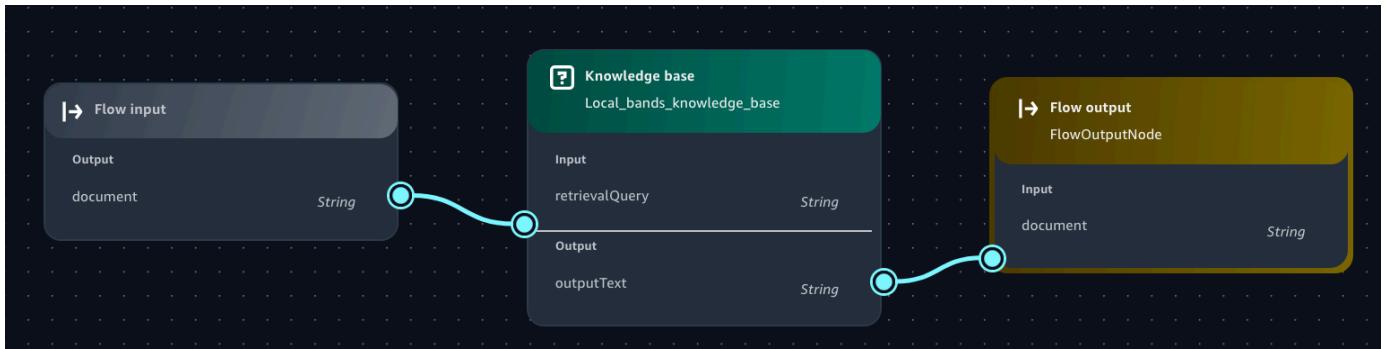
To add a Knowledge Base component to a chat agent app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. In **Apps** choose the chat agent app that you want to add the knowledge base component to.
7. In the **Configs** pane, choose **Data**.
8. Select **Use Knowledge Base**.
9. For **Select Knowledge Base**, select the Knowledge Base component that you want to use. To create a Knowledge Base component, see [Create an Amazon Bedrock Knowledge Base component](#).
10. (Optional) Choose **Edit advanced search configs** to set advanced search configurations.
 - a. In **Search type**, turn on **Override default search** to choose a different search type. You can choose from **Hybrid search** (Combines relevancy scores from semantic and text search to provide greater accuracy) or **Semantic search** (Uses vector embeddings to deliver relevant results).
 - b. (Optional) In **Maximum number of source chunks**, choose the maximum number of source chunks to use.
11. Choose **Save** to save your changes.

Add a Knowledge Base component to a flow app

In this procedure, you add a Knowledge Base component to an existing [flow app](#).

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. In **Apps** choose the flow app that you want to add the knowledge base component to.
7. In the **Flow app builder** pane, select the **Nodes** tab.
8. From the **Data** section, drag a **Knowledge Base** node onto the flow builder canvas.
9. The circles on the nodes are connection points. Draw a line from the circle on the upstream node (such as the **Flow input** node) to the circle on the **Input** section of the Knowledge Base node that you just added.
10. Connect the **Output** of the Knowledge Base node to the downstream node that you want the Knowledge Base to send its output to. The flow should look similar to the following image:



11. In the the flow builder, select the Knowledge Base node that you just added.
12. In the **flow builder** pane, choose the **Configure** tab and do the following:
 - a. For **Node name**, enter a name for the Knowledge Base node.
 - b. For **Select Knowledge Base** in the **Knowledge Base Details** section, select the Knowledge Base that you just created.

- c. For **Select response generation model**, select the model that you want the Knowledge Base to generate responses with.
 - d. (Optional) In **Select guardrail** select an existing guardrail. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).
13. Choose **Save** to save your changes.

Synchronize an Amazon Bedrock Knowledge Base

After you create a Knowledge Base data source, you synchronize your data so that the data can be queried. Synchronization converts the raw data in your data source into vector embeddings, based on the vector embeddings model and configurations you specified when you [Created](#) the Knowledge Base.

If the data source is a web crawler, synchronization time can vary from minutes to hours, depending on the URLs you define.

To synchronize a Knowledge Base

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.
5. In the **Select or create a new project to continue** dialog box, select the project that you want to use.
6. In the left pane, choose **Asset gallery**.
7. In **Asset gallery**, choose **My components**.
8. Find the Knowledge Base that you want to synchronize, and choose the menu option and select **Sync**.
9. Wait until the Knowledge Synchronization completes.

Safeguard your Amazon Bedrock app with a guardrail

Guardrails for Amazon Bedrock lets you implement safeguards for your Amazon Bedrock in SageMaker Unified Studio app based on your use cases and responsible AI policies. You can create multiple guardrails tailored to different use cases and apply them across multiple foundation models, providing a consistent user experience and standardizing safety controls across generative AI apps. You can configure denied topics to disallow undesirable topics and content filters to block harmful content in the prompts you send to a model and to the responses you get from a model. You can use guardrails with text-only foundation models. For more information, see [Safeguard your Amazon Bedrock app with a guardrail](#).

You can use guardrails with Amazon Bedrock in SageMaker Unified Studio chat agent app and with flow apps. With a chat agent app you can create guardrail component when you [create the chat agent app](#) or you can add a guardrail component that you have previously created. For more information, see [Create an Amazon Bedrock guardrail component](#).

With a flow app, you can add a guardrail to [prompt](#) nodes and to [knowledge base](#) nodes.

Topics

- [Guardrail policies](#)
- [Create an Amazon Bedrock guardrail component](#)
- [Add an Amazon Bedrock guardrail component to a chat agent app](#)
- [Add an Amazon Bedrock guardrail component to a flow app](#)

Guardrail policies

A guardrail consists of the following policies to avoid content that falls into undesirable or harmful categories.

- Content filters – Adjust filter strengths to filter input prompts or model responses containing harmful content.
- Denied topics – You can define a set of topics that are undesirable in the context of your app. These topics will be blocked if detected in user queries or model responses.

Content filters

Guardrails in Amazon Bedrock in SageMaker Unified Studio support the following content filters to detect and filter harmful user inputs and FM-generated outputs.

- **Hate** – Describes language or a statement that discriminates, criticizes, insults, denounces, or dehumanizes a person or group on the basis of an identity (such as race, ethnicity, gender, religion, sexual orientation, ability, and national origin).
- **Insults** – Describes language or a statement that includes demeaning, humiliating, mocking, insulting, or belittling language. This type of language is also labeled as bullying.
- **Sexual** – Describes language or a statement that indicates sexual interest, activity, or arousal using direct or indirect references to body parts, physical traits, or sex.
- **Violence** – Describes language or a statement that includes glorification of or threats to inflict physical pain, hurt, or injury toward a person, group or thing.

Content filtering depends on the confidence classification of user inputs and FM responses across each of the four harmful categories. All input and output statements are classified into one of four confidence levels (NONE, LOW, MEDIUM, HIGH) for each harmful category. For example, if a statement is classified as *Hate* with HIGH confidence, the likelihood of the statement representing hateful content is high. A single statement can be classified across multiple categories with varying confidence levels. For example, a single statement can be classified as *Hate* with HIGH confidence, *Insults* with LOW confidence, *Sexual* with NONE confidence, and *Violence* with MEDIUM confidence.

For each of the harmful categories, you can configure the strength of the filters. The filter strength determines the degree of filtering harmful content. As you increase the filter strength, the likelihood of filtering harmful content increases and the probability of seeing harmful content in your app reduces. The following table shows the degree of content that each filter strength blocks and allows.

Filter strength	Blocked content confidence	Allowed content confidence
None	No filtering	None, Low, Medium, High
Low	High	None, Low, Medium
Medium	High, Medium	None, Low

Filter strength	Blocked content confidence	Allowed content confidence
High	High, Medium, Low	None

Denied topics

Guardrails can be configured with a set of denied topics that are undesirable in the context of your generative AI app. For example, a bank may want their online assistant to avoid any conversation related to investment advice or engage in conversations related to fraudulent activities such as money laundering.

You can define up to five denied topics. Input prompts and model completions will be evaluated against each of these topics. If one of the topics is detected, the blocked message configured as part of the guardrail will be returned to the user.

Denied topics can be defined by providing a natural language definition of the topic along with a few optional example phrases of the topic. The definition and example phrases are used to detect if an input prompt or a model completion belongs to the topic.

Denied topics are defined with the following parameters.

- Name – The name of the topic. The name should be a noun phrase. Don't describe the topic in the name. For example:
 - **Investment Advice**
- Definition – Up to 200 characters summarizing the topic content. The description should describe the content of the topic and its subtopics.

 **Note**

For best results, adhere to the following principles:

- Don't include examples or instructions in the description.
- Don't use negative language (such as "don't talk about investment" or "no content about investment").

The following is an example topic description that you can provide:

- **Investment advice refers to inquiries, guidance or recommendations regarding the management or allocation of funds or assets with the goal of generating returns or achieving specific financial objectives.**
- Sample phrases – A list of up to five sample phrases that refer to the topic. Each phrase can be up to 1,000 characters. An sample is a prompt or continuation that shows what kind of content should be filtered out. For example:
 - **Is investing in the stocks better than bonds?**
 - **Should I invest in gold?**

Create an Amazon Bedrock guardrail component

You can create a guardrail as a component in an Amazon Bedrock in SageMaker Unified Studio project. You can then add the guardrail component to a chat agent app. You can also create a guardrail component while you are creating a chat agent app. For an example, see [Step 2: Add a guardrail to your chat agent app](#).

To create a guardrail component

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.
5. In the **Select or create a new project to continue** dialog box, select the project that you want to use.
6. In the left pane, choose **Asset gallery**.
7. Choose **My components**.
8. In the **Components** section, choose **Create component** and then **Guardrail**. The **Create guardrail** pane is shown.
9. For **Guardrail name**, enter a name for the guardrail.
10. For **Guardrail description** enter a description for the guardrail.
11. In **Content filters** do the following.

- a. Select **Enable content filters** to turn on content filtering.
 - b. For **Filter for prompts**, choose the filters that you want to apply to prompts. For more information, see [Content filters](#).
 - c. If you want the filter to apply to responses that the model generates, select **Apply the same filters for responses**.
12. In **Blocked messaging** do the following:
- a. For **Blocked messaging for prompts** enter a message to display when the guardrail blocks content in the prompt.
 - b. If you want to show a different message when the guardrail blocks content from a model's response, do the following:
 - i. Clear **Apply the same message for blocked responses**.
 - ii. For **Blocked messaging for responses**, enter a message to display when the guardrail blocks content in the response from the model.
13. Add a denied topic filter by doing the following:
- a. Choose **Use advanced features**.
 - b. Choose **Denied topics**.
 - c. Choose **Add topic**.
 - d. For **Name**, enter a name for the filter.
 - e. For **Definition for topic**, enter a definition for the content that you want to deny.
 - f. (Optional) To help guide the guardrail, do the following:
 - i. Choose **Sample phrases - optional**.
 - ii. For **Sample phrases**, enter a phrase.
 - iii. Choose **Add phrase**.
 - iv. Add up to four more phrases by repeating the previous two steps.
 - v. Choose **Save**.

For information about denied topics, see [Denied topics](#).

14. Choose **Create** to create the guardrail.

15. Add the guardrail component to a chat agent app by doing [Add an Amazon Bedrock guardrail component to a chat agent app](#).

Add an Amazon Bedrock guardrail component to a chat agent app

In this procedure, you add a guardrail component to an existing [chat agent app](#).

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. In **Apps** choose the chat agent app that you want to add the guardrail to.
7. In the **Configs** pane, choose **Guardrails**.
8. For **Guardrails**, select the guardrail component that you created in [Create an Amazon Bedrock guardrail component](#).
9. (Optional) Preview the guardail by choosing **Preview**. From the preview you can edit the guardrail, if desired.
10. Choose **Save** to save your changes.

Add an Amazon Bedrock guardrail component to a flow app

In this procedure, you add a guardrail component to the [knowledge base](#) node that you create in step 2 of [Create a flow app with Amazon Bedrock](#). You can also add a guardrail to a [prompt](#) node. To add a guardrail to a prompt node, use the following steps. For step 7, select the prompt node (**Playlist_generator_node**).

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. open the flow app that you created in [Create a flow app with Amazon Bedrock](#).
7. In the canvas, select the knowledge base node (**Local_bands_knowledge_base**).
8. In the **flow builder** pane, choose the configure tab.
9. In the **Guardrail details** section do one of the following:
 - Select an existing guardrail to use
 - Choose **Create a new guardrail** to create a new guardrail. Then do the following:
 1. For **Guardrail name**, enter a name for the guardrail.
 2. For **Guardrail description** enter a description for the guardrail.
 3. In **Content filters** do the following:
 - a. Select **Enable content filters** to turn on content filtering.
 - b. For **Filter for prompts**, choose the filters that you want to apply to prompts. For more information, see [Content filters](#).
 - c. If you want the filter to apply to responses that the model generates, select **Apply the same filters for responses**.
 4. In **Blocked messaging** do the following:
 - a. For **Blocked messaging for prompts** enter a message to display when the guardrail blocks content in the prompt.
 - b. If you want to show a different message when the guardrail blocks content, do the following:

- i. Clear **Apply the same message for blocked responses.**
 - ii. For **Blocked messaging for responses**, enter a message to display when the guardrail blocks content in the response from the model.
5. Add a denied topic filter by doing the following:
- a. Choose **Use advanced features**.
 - b. Choose **Denied topics**.
 - c. Choose **Add topic**.
 - d. For **Name**, enter a name for the filter.
 - e. For **Definition for topic**, enter a definition for the content that you want to deny.
 - f. (Optional) To help guide the guardrail, do the following:
 - i. Choose **Sample phrases - optional**.
 - ii. For **Sample phrases**, enter a phrase.
 - iii. Choose **Add phrase**.
 - iv. Add up to four more phrases by repeating the previous two steps.
 - v. Choose **Save**.

For information about denied topics, see [Denied topics](#).

6. Choose **Create** to create the guardrail.
 7. Add the guardrail component to a chat agent app by doing [Add an Amazon Bedrock guardrail component to a chat agent app](#).
10. Choose **Save** to save your changes to your flow app.
11. Test your prompt by doing the following:
- a. On right side of the app flow page, choose < to open the test pane.
 - b. In **Enter prompt**, enter a phrase that violates your guardrail.
 - c. Press Enter on the keyboard or choose the run button to test the prompt.
 - d. In the response you should see the text **guardrail applied** under the affected prompt. Choose the prompt to see the reason why the guardrail rejected the prompt.

Call functions from your Amazon Bedrock chat agent app

Amazon Bedrock in SageMaker Unified Studio functions let a model include information that it has no previous knowledge of in its response. For example, you can use a function to include dynamic information in a model's response such as a weather forecast, sports results, or traffic conditions.

In Amazon Bedrock in SageMaker Unified Studio, a function calls an API hosted outside of Amazon Bedrock in SageMaker Unified Studio. You either create the API yourself, or use an existing API. To create an API, you can use [Amazon API Gateway](#).

To use a function in Amazon Bedrock in SageMaker Unified Studio you add a *function component* to your app. As part of the function, you define an OpenAPI schema for the API that you want the model to call. You also specify how to authenticate the call to the API. When a model receives a prompt, it uses the schema and the prompt to determine if an API should be called and the parameters that the API should receive. If the API is called, the response from the model includes the output from the API.

APIs that you call in a function must have a response size that is less than 20K.

When add a function to an app, you need to specify the app's system instruction. The system instruction needs to be at least 40 characters long and should mention the new skills that the new function introduces.

You can use functions in a [chat agent app](#).

Topics

- [Function schema](#)
- [Authentication methods](#)
- [Create an Amazon Bedrock function component](#)
- [Add a function component to an Amazon Bedrock chat agent app](#)

Function schema

Amazon Bedrock in SageMaker Unified Studio has the following requirements for the schema that you use to create a function.

- The function schema must be [OpenAPI version 3.0.0](#).

- The function schema must be in JSON or YAML format.
- The function can have no authentication, API key authentication, Bearer token authentication, or basic authentication. For more information, see [Authentication methods](#).
- You can have 0 or 1 server URL.
- All [Operation Objects](#) must have a description.
- All [Parameter Objects](#) must have a description.
- [Security scheme object](#) must have a type that is either apiKey or http.

When the type is http, the scheme field must either be basic or bearer.

When the type is apiKey, the in property must be query or header. Also, the name property must be defined.

- Amazon Bedrock in SageMaker Unified Studio only honors [globally-scoped security requirement](#). For more information, see [Valid components for globally-scoped security requirements](#).
- Parameters (**parameter.in**) must be passed through query or path. You can't use cookies or headers to pass parameters.
- Parameters (**parameter schema type**) must be primitive types, arrays, or objects (one-level JSON). You can't pass complex nested objects.
- Parameter content (**parameter.content**) is mutually exclusive with the schema. Schema is more commonly used. Use content only for more complex types, or for complex serialization scenarios that are not covered by style and explode.
- Parameter **style** and **explode** values. form and true for query, simple and false for paths). For more information, see [Parameter Serialization](#).
- Request body content must be passed as application/json.
- The schema can have up to 5 APIs and an app can use up to 5 APIs across all functions. For the model to correctly choose function, it is important to provide detailed descriptions of the API, including parameters, properties, and responses.

Valid components for globally-scoped security requirements

Amazon Bedrock in SageMaker Unified Studio only honors [globally-scoped security requirements](#). That is, Amazon Bedrock in SageMaker Unified Studio ignores security requirements indicated in operation objects.

When the requirement array contains a security scheme object with type `http` and scheme of `bearer` or `basic`, the array must contain a single entry. Amazon Bedrock in SageMaker Unified Studio ignores further entries.

When the requirement array contains a security scheme object with type `apiKey`, you can have a maximum of 2 entries.

For example, if you have the following [components](#):

```
"components": {  
    "securitySchemes": {  
        "api_key_1": {  
            "type": "apiKey",  
            "name": "appid1",  
            "in": "query"  
        },  
        "api_key_2": {  
            "type": "apiKey",  
            "name": "appid2",  
            "in": "header"  
        },  
        "api_key_3": {  
            "type": "apiKey",  
            "name": "appid3",  
            "in": "cookie"  
        },  
        "bearer_1": {  
            "type": "http",  
            "scheme": "bearer",  
        },  
        "bearer_2": {  
            "type": "http",  
            "scheme": "bearer",  
        },  
        "basic_1": {  
            "type": "http",  
            "scheme": "basic",  
        },  
        "basic_2": {  
            "type": "http",  
            "scheme": "basic",  
        },  
        "http_digest": {  
    }  
}
```

```
        "type": "http",
        "scheme": "digest"
    },
    "oauth2_1": {
        "type": "oauth2"
    }
}
```

The following are valid:

```
# 1 API key
"security": [
{
    "api_key_1": []
}
],  
  
# 2 API keys
"security": {
{
    "api_key_1": [],
    "api_key_2": []
}
}  
  
# Bearer
"security": {
    "bearer_1": []
}  
  
# Basic
"security": {
    "basic_1": []
}
```

The following are invalid:

```
# Invalid: `type` must only be `apiKey` or `http`
"security": {
    "oauth2_1": []
}
```

```
# Invalid: `scheme` must only be `basic` or `bearer` if `type` is `http`
"security": {
    "http_digest": []
}

# Invalid: `security` must only contain 1 entry if `type` is `basic` or `bearer`
"security": {
    "basic_1": [],
    "basic_2": []
}

# Invalid: `security` must not contain varying security types
"security": {
    "api_key_1": [],
    "basic_1": []
}

# Invalid: API key must only have `in` property set to `header` or `query`
"security": {
    "api_key_1": [],
    "api_key_3": []
}

# Invalid: `security` must not have more than 2 API keys
"security": {
    {
        "api_key_1": [],
        "api_key_2": [],
        "api_key_3": []
    }
}
```

Authentication methods

Amazon Bedrock in SageMaker Unified Studio supports the following methods for authenticating function calls to an API server. If you authenticate a function call, make sure the credentials you provide are correct as Amazon Bedrock in SageMaker Unified Studio doesn't verify the credentials before you use them in a function call.

- **No authentication** – No authentication means that the client doesn't need to provide any credentials to access a resource or service. This method is typically used for publicly available resources that don't require any form of authentication.

- **API keys** – An API key is a unique identifier used to authenticate a client application and allow it to access an API or service. You can add a maximum of two keys.
- **Bearer token** – A bearer token is an opaque string that represents an authentication credential. It is typically obtained after a successful authentication process, such as OAuth 2.0. This method allows the client to access protected resources without having to send the actual credentials (username and password) with each request.

 **Note**

Amazon Bedrock in SageMaker Unified Studio is unable to assure whether the token is valid or has already expired. It is your responsibility to make sure that you provide a valid token, and to update the token to a new one before it expires. If the token expires, Amazon Bedrock won't be able to successfully call APIs with the token.

- **Basic authentication** – Basic authentication is a simple authentication scheme built into the HTTP protocol. The credentials are sent with every request, which can be a security concern if the connection is not secured using HTTPS. Basic authentication is generally considered less secure than other modern authentication methods and should be used with caution, especially in production environments.

Create an Amazon Bedrock function component

You can create a function as a component in an Amazon Bedrock in SageMaker Unified Studio project. If you are creating an app, you can also create a function when you configure the app.

To create a function component

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called “Access Amazon SageMaker Unified Studio”](#).
3. Choose the **Build** menu at the top of the page.
4. In the **MACHINE LEARNING & GENERATIVE AI** section, choose **My apps**.
5. In the **Select or create a new project to continue** dialog box, select the project that you want to use.
6. In the left pane, choose **Asset gallery**.

7. Choose **My components**.
8. In **Asset gallery**, choose **My components**.
9. In the **Components** section, choose **Create component** and then **Function**. The **Create function** pane is shown.
10. For **Function name**, enter a name for the function in **Function name**.
11. For **Function description**, enter a description for the function.
12. For **Function schema**, enter the JSON or YAML format OpenAPI schema for the API. Alternatively, upload the JSON or YAML for the file by choosing **Import JSON/YAML**. You can clear the text box by choosing **Reset**.
13. Choose **Validate schema** to validate the schema.
14. For **Authentication method** select the authentication method for your API server. By default, Amazon Bedrock in SageMaker Unified Studio preselects the authentication based on information it finds in your OpenAPI schema. For information about authentication methods, see [Authentication methods](#).
15. Enter the information for the authentication method that you selected in the previous step.
16. For **API servers**, enter the URL for your server in **Server URL**. This value is autopopulated if the server URL is in the schema.
17. Choose **Create** to create your function.
18. Add your function to a chat agent app by doing [Add a function component to an Amazon Bedrock chat agent app](#).

Add a function component to an Amazon Bedrock chat agent app

In this procedure, you add a function component to an existing [chat agent app](#). You can add up to 5 functions to an app. For each function you add, be sure to update the system instruction with information about the function.

To add a function component to a chat agent app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. If the project that you want to use isn't already open, do the following:

- a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
 5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
 6. In **Apps** choose the chat agent app that you want to add the function component to.
 7. In the **Configs** pane, do the following:
 - a. For **Enter a system instruction**, enter or update the system prompt so that it describes the function.
 - b. Choose **Functions**.
 - c. For **Functions**, select the function component that you created in [Create an Amazon Bedrock function component](#).
 8. Choose **Save** to save your changes.

Use app history to view and restore versions of an Amazon Bedrock app

As you develop an Amazon Bedrock in SageMaker Unified Studio app (chat agent app or flow app), you make changes and improvements. When you save an app, Amazon Bedrock in SageMaker Unified Studio saves the current draft of the app (including its configuration information), as a version in the *app history*. At some point, you might want to view and restore a previous version of an app. For example, you might want to check the guardrail that a previous version of a chat agent app uses, or you might want continue development starting from a previous version of the app. You can use the app history to view and restore previous app versions.

Within the app history, you can restore a previous version of the app. If you don't save the current draft before restoring a previous version, Amazon Bedrock in SageMaker Unified Studio automatically saves the draft for you. While you are viewing the app history for an app, you can't make changes to the app.

To view or restore a previous version of an app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. In **Apps** choose the app that you want to use.
7. On the **Save** button, choose the menu selector and then select **View history**.
8. In the **History pane**, select the version of the app that you want to view. The center pane refreshes to contain the selected app version.
9. In the center pane, view the app version and its configuration. You can't make changes to app. To return to the editable draft, choose **Close App history**.
10. (Optional) In the **App history** pane, choose **Restore** to restore the app version. After restoring the app, Amazon Bedrock in SageMaker Unified Studio closes the app history pane and you can edit the restored app version.

If you don't save the current draft before restoring a previous version, Amazon Bedrock in SageMaker Unified Studio automatically saves the draft for you.

Use your app outside of Amazon SageMaker Unified Studio

With Amazon Bedrock in SageMaker Unified Studio, you can export the files for an [chat agent app](#) and a [flow app](#). This lets you use the app outside of Amazon SageMaker Unified Studio.

When you export an app, Amazon Bedrock in SageMaker Unified Studio exports a zip file with the AWS CloudFormation templates and other files required by your app. To use your app, you need to deploy the AWS CloudFormation templates to an AWS account. The actual contents of the zip

file vary on the Amazon Bedrock in SageMaker Unified Studio components that your app uses. After uncompressing the zip file, you deploy the contents of the zip file into your AWS account (or another AWS account, if you prefer).

Important

Once you export your app, it's your responsibility to audit the app files and make sure they are correct. You can use the AWS CloudFormation templates as you wish.

An app can include one or more different types of Amazon Bedrock in SageMaker Unified Studio components. For example, a chat agent app could use a guardrail or a knowledge base. When you deploy your app's components, Amazon Bedrock in SageMaker Unified Studio only deploys the AWS infrastructure files. The data source files for a knowledge base and the secrets for a function aren't exported, and you have to configure them during the deployment. After deploying the app to an AWS account, you can run the app as a Node.js app.

App export files

Depending on the composition of your app, the zip package contains some or all of the following files:

- **README.md** — Instructions for deploying and running your app.
- **function-stack-***.json**** — AWS CloudFormation template that creates your function component, if any. This includes:
 - An AWS Lambda [function](#) for calling the API defined in your OpenAPI schema.
 - An AWS Secret Manager secret for storing credentials to use when calling your API. This secret contains an empty value, and you are expected to update this secret manually.
- **knowledge-base-stack-***.json**** — AWS CloudFormation template that creates your [Knowledge Base data source](#), if any. This includes an Knowledge Base for Amazon Bedrock configured with your selected data store and vector store. This knowledge base will not have the data you have uploaded in to Amazon Bedrock in SageMaker Unified Studio, and you are expected to provide data files manually.
- **flow-stack.json** — AWS CloudFormation template that creates an Amazon Bedrock flows resource.
- **guardrails-stack-***.json**** — AWS CloudFormation template that creates a [guardrail](#) for Amazon Bedrock, if any.

- **agent-stack.json** — AWS CloudFormation template that creates an Amazon Bedrock Agent, if any.
- **invocation-policy-*json** — AWS CloudFormation template that creates an IAM policy with the runtime permissions that you need to talk to your deployed chat agent app.
- **br-studio-app-stack-*json** — Parent stack that orchestrates the deployment of all AWS CloudFormation stacks included in the zip package.
- **deploy-app.sh** — Helper script that you use to deploy your app infrastructure into your AWS account.
- **code-snippet.mjs** — Example code snippet that you embed in your code to invoke the app.
- **amazon-bedrock-ide-app.mjs** — Standalone Node.js module to quickly test your deployed app.
- **aoss-encryption-policy-*json** — AOSS encryption policy necessary to use a Knowledge Base. This encryption policy is automatically created when your chat agent app contains an Amazon Bedrock in SageMaker Unified Studio Knowledge Base.
- **provisioning-inline-policy.json** — An example of an AWS IAM policy that contains the permissions required to provision the chat agent app resources. The permissions declared in this policy file are needed when deploying the AWS CloudFormation stacks.

You can modify this policy to better suit your needs. You may create a new IAM principal with these policies, or attach these policies to an existing IAM principal in your AWS account.

- **kms-key-policy.json** — An example of an AWS KMS key policy that contains required permissions for encrypting your chat agent app resources.

You can modify this key policy to better suit your needs. You may create a new KMS key with this policy, or attach this policy to an existing KMS key in your AWS account.

- **api-schema*.json** — OpenAPI schema files associated with your function components, if any.

Topics

- [Export your Amazon Bedrock app](#)
- [Deploy an exported Amazon Bedrock app](#)
- [Run a deployed Amazon Bedrock app](#)

Export your Amazon Bedrock app

Use the following procedure to export a chat agent app or a flow app to a zip file. You can then use the app outside of Amazon SageMaker Unified Studio.

To export a chat agent app or a flow app

1. Navigate to the Amazon SageMaker Unified Studio landing page by using the URL from your administrator.
2. Access Amazon SageMaker Unified Studio using your IAM or single sign-on (SSO) credentials. For more information, see [the section called "Access Amazon SageMaker Unified Studio"](#).
3. If the project that you want to use isn't already open, do the following:
 - a. Choose the current project at the top of the page. If a project isn't already open, choose **Select a project**.
 - b. Select **Browse all projects**.
 - c. In **Projects** select the project that you want to use.
4. Choose the **Build** menu option at the top of the page.
5. In **MACHINE LEARNING & GENERATIVE AI** choose **My apps**.
6. In **Apps** choose the app that you want to export.
7. If you haven't already, choose **Save** to save the app. You can't export an app unless you first save and run the app.
8. On the app page, choose **Export** to export the app. Amazon Bedrock in SageMaker Unified Studio will create and download a zip file with the name **amazon-bedrock-ide-app-export-*zip**.
9. Next step: [Deploy the app](#).

Deploy an exported Amazon Bedrock app

The following instructions show you the steps you take to deploy a chat agent app that you [export](#) from Amazon Bedrock in SageMaker Unified Studio. Make sure to

Topics

- [Prerequisites for deploying an exported app](#)
- [Deploy the exported app](#)

Prerequisites for deploying an exported app

Before you can deploy a chat agent app that you have exported, you must first do the following:

To prepare for app deployment

1. Install the latest version of the AWS CLI on your local machine by following the instructions at [Install or update to the latest version of the AWS CLI](#).
2. Set up AWS credentials for the AWS CLI on your local machine by following the instructions at [Configure the AWS CLI](#). The credentials that the deployment script uses will follow the [order of precedence](#).
3. (Optional) Using the AWS account that you set up in step 2, create an AWS KMS key for app export by following the instructions at [Creating keys](#). The key must be tagged with key EnableBedrock and a value of true. The key must also have a key policy that allows it to be used for encryption of your chat agent app resources. You may use the suggested policy declared in the kms-key-policy.json file of your zip package.
4. Create an Amazon S3 bucket to hold the app files that you export by following the instructions at [Creating a bucket](#). Make sure the bucket is in the same AWS Region as the app that you are deploying.
5. Create an IAM role that includes the policies from provisioning-inline-policy.json. For information about creating a role, see [IAM role creation](#).
6. If your app includes a Knowledge Base, copy the data source file to a folder named data/ in the Amazon S3 bucket that you created in step 4. If your app uses a document as a datasource, you supply a list of datasource files to the deployment script. For more information, see [Deploy the exported app](#).
7. If your app calls a function that requires authorization, update the function environment secret in Amazon SageMaker AI to the authorization method that your function uses. Run the following command:

```
aws secretsmanager update-secret \
--secret-id br-studio/function-name-export-environment-id \
--secret-string 'secret-value'
```

To get the *function-name* and *export-environment-id* values, open the *amazon-bedrock-ide-app-stack-nnnn.json* file from the files that you exported in [Export your Amazon Bedrock app](#). The values are in the FunctionsStack0 JSON object.

Replace the following values:

- **function-name** — to the value of the `functionName` field in the `FunctionsStack0` JSON object.
- **export-environment-id** — to the value of the `exportAppInstanceId` field in the `FunctionsStack0` JSON object.
- **secret-value** — to the intended value to be used for authentication. You specified the authentication type when you [created the function component](#). Use the authentication values that you specified to complete the `secret-value`.

If the function requires API Keys, the syntax of `secret-value` should be: `{"key-name-1": "key-value-1", "key-name-2": "key-value-2"}`

If the function requires Basic authentication, the syntax of `secret-value` should be: `{"__AuthType__": "BASIC", "username": "username-value", "password": "password-value"}`

If the function requires Bearer token authentication, the syntax of `secret-value` should be: `{"__AuthType__": "BEARER", "tokenValue": "token-value"}`

8. Next step: [Deploy the exported app](#).

Deploy the exported app

Before deploying your chat agent app, be sure to do the [prerequisite steps](#).

Deploying a chat agent app deploys the AWS infrastructure files that you need to run the app in AWS.

To deploy an exported app

1. At the command prompt, do the following:
 - a. Navigate to the zip file that you exported from Amazon Bedrock in SageMaker Unified Studio.
 - b. Assume the role of the AWS that you created in step 3 of [Prerequisites for deploying an exported app](#).

- c. Use the following command to make sure the deployment script (`deployApp.sh`) is executable:

```
chmod +x deployApp.sh
```

- d. Run the deployment script with the following command:

```
./deployApp.sh \
[--awsRegion=value] \
[--s3BucketName=value] \
[--assetsS3Path=value] \
[--kmsKeyArn=value] \
[--dataFiles=value]
```

Replace the following values:

- `awsRegion` — with the AWS Region that you want to deploy the app to. Amazon Bedrock must be available in the Region you use. For more information, see [Supported AWS Regions](#).
- `s3BucketName` — With the Amazon S3 bucket that you created in step 5 of [Prerequisites for deploying an exported app](#). The deployment store the CFN templates and application data files in this bucket.
- `assetsS3Path` — (Optional) With the path in `s3BucketName` that you want deployment to store application files to.
- `kmsKeyArn` — (Optional) with the ARN of the KMS Key that you created in step 3 of [Prerequisites for deploying an exported app](#).
- `dataFiles` — With a comma-separated list of data source file paths. Required for apps that use a document data source.

For example, if you have a chat agent app with a single document as a data source, and you want to deploy the app with encryption, you can use the following command.

```
./deployApp.sh \
--awsRegion=us-east-1 \
--s3BucketName=my-s3-bucket-name-for-exported-chat-apps \
--assetsS3Path=my-prod-folder/my-chat-app \
--kmsKeyArn=arn:aws:kms:us-
east-1:111122223333:key/11111111-2222-3333-4444-555555555555 \
```

```
--dataFiles=my-data-source.pdf
```

2. (Optional) Monitor the deployment in the AWS CloudFormation console.
3. Note the output from the script. You need it to run the chat agent app. It should be similar to: node amazon-bedrock-ide-app.mjs --question="*prompt*" --region="*AWS Region*".

When you run the app, specify the following parameters:

- **question** – The prompt that you want to start the app with.
- **region** – The AWS Region that you deployed the app to. Use the value of `awsRegion` that you specified in step 1c.

For example, `node amazon-bedrock-ide-app.mjs --question="Tell me about my documents" --region="us-east-1"`

4. Next step: [Run a deployed Amazon Bedrock app](#).

Run a deployed Amazon Bedrock app

The following instructions show you the steps you take to run a deployed Amazon Bedrock in SageMaker Unified Studio chat agent app.

Topics

- [Prerequisites for running a chat agent app](#)
- [Run the app](#)

Prerequisites for running a chat agent app

Before you can run an app that you have exported, you must first do the following:

To prepare for running an app

1. Download and install Node.js. For more information, see [Download Node.js](#).
2. At the command prompt, install third-party Node.js libraries by running the following commands:

```
npm install minimist
```

```
npm install aws-sdk
npm install @aws-sdk/credential-providers
npm install @aws-sdk/client-bedrock-agent-runtime
npm install @aws-sdk/client-bedrock-runtime
```

For a flow app you also need the following

```
npm install @aws-sdk/client-bedrock-agent
```

3. Create or update an IAM role in which you want to run the app. For the policy, use the policy created by `deployApp.sh` when you exported the app. The policy name is `BRStudioExportedAppInvocationRolePolicy-exportProjectId`. The policy is declared in `invocation-policy-* .json`. For more information, see [Creating roles](#).

Run the app

To run your app, you need an IAM role with permissions to invoke Amazon Bedrock resources. When you deploy the app, the AWS CloudFormation stack deployed through `deployApp.sh` script provisions a suitable policy in your AWS account (declared in `invocation-policy-* .json`).

To run the app

1. Switch to the IAM role that you created in step 3 of [Prerequisites for running a chat agent app](#).
2. Run the app by entering the command you noted in step 3 of [Deploy the exported app](#).

SQL analytics

You can use the query editor to perform analysis using SQL. The query editor tool provides a place to write and run queries, view results, and share your work with your team.

For information about getting started with the query editor, see [the section called “Get started with the query editor”](#).

Topics

- [Navigate the query editor](#)
- [Connect data resources](#)
- [Supported query engines](#)
- [Create a query](#)
- [Generative SQL](#)
- [Review query history](#)
- [Schedule a query](#)

Navigate the query editor

The query editor lets you run queries on your data. You can work directly with database objects and manipulate them through SQL statements. The query editor is primarily used to edit and run queries, visualize results, and share your work with your team. Once a data cluster has been added to the project, you can see the data in the navigation tree on the left.

There are three options that you can use to navigate in the Query Editor:

- The data explorer icon displays all data assets that you can query.



Using the dataset icon, you can use the three dots next to the name of a table to run a sample query. When you choose **Query with Athena** or **Query with Redshift**, the query runs and displays results.

- The file explorer icon displays all the querybooks that you or other project members have saved to the project repository.



- The query history icon displays past queries that you have run.



Connect data resources

You can add a data source to the query editor by using the + icon on the data explorer page. There are three different ways you can add a data source:

- Add a connection to an existing data source.
- Upload data from your computer.
- Create a new catalog for your Amazon Redshift managed storage objects.

Supported query engines

The Amazon SageMaker Unified Studio query editor supports the following query engines:

- Amazon Redshift. For more information, see [Query processing](#) in the Amazon Amazon Redshift Database Developer Guide.
- Amazon Athena. For more information, see [Running SQL queries using Amazon Athena](#) in the Amazon Amazon Athena User Guide.

Both engines use Querybooks to develop queries and work with data from one place. You can change the query engine from the upper-right corner of the Querybook editor and selecting the data source you want to use from the dropdown menu.

Create a query

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to a project that uses the **SQL analytics** or **Data analytics and AI-ML model development** project profile. You can do this by using the center menu at the top of the page

and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.

3. In the **Build** menu, choose **Query editor**. This takes you to the Query editor page.
4. To start writing a query, choose the + icon at the top of the editor window to create a new tab. This creates a new querybook with an empty SQL cell.
5. Write the query in the given SQL cell. For a guide to SQL commands you can use, see [SQL reference](#) in the Amazon Redshift Database Developer Guide or the [SQL reference](#) for Amazon Athena.
6. (Optional) To create more space for writing commands, add another SQL or markdown cell to the notebook.

To run a query, choose the **Run** icon to run queries in a specific cell, or choose **Run all** at the top of the editor window to run all queries on the page. A query runs until it is finished or until you use the **Stop** button.

After running a query, save the SQLNB file to the project to share it with other project members. The steps are as follows:

1. Expand the **Actions** menu and choose **Save to project**.
2. Confirm the action by choosing **Save changes**.

Generative SQL

Amazon SageMaker Unified Studio allows you to create SQL queries by interacting with a chatbot powered by Amazon Q. The text-to-SQL interpreter understands the context of your data and generates SQL queries that you can use in your analysis.

You can find the Generative SQL tool in the Query Editor of a project.

The screenshot shows the Amazon SageMaker Unified Studio interface. On the left, the Query Editor window is open, displaying a single cell with the number '1' and the text 'Row 1, Col 1, Chr 0'. Below the cell are buttons for 'Add SQL' and 'Add markdown'. At the bottom of the editor are keyboard shortcuts: '(Cntrl+Enter, Cmd+Enter) to run', '(Shift+Enter) to add new cell', and '(Cntrl+M, Cmd+M) to add new markdown'. On the right, the Amazon Q interface is shown. It includes a header 'Amazon Q' with icons for profile, help, and close. Below the header is a text area: 'I am a generative SQL assistant, your coding companion that generates SQL statements based on your database.' followed by 'You can add custom context that describes your data and queries and use it to generate accurate SQL. Your data remains private to your account. To add custom context, click settings.' A link 'For more information, see Interacting with query editor generative SQL' is provided. Further down, there's a section 'Connect to a database and enter your English natural language query at the prompt.' and 'For examples using the tickit data' with two examples: 'Which state has most venues' and 'Find the top five users from Seattle who bought the most number of tickets'. At the bottom of the Amazon Q interface is a text input field 'Ask me a question to generate a SQL suggestion for your connected database.' with a character count indicator '500 characters left' and a note about AI developer usage.

The Generative SQL chat application understands the context of your data and will let you ask natural language questions like "What are the columns in the job success table?", in order to generate SQL. When you ask a question, an SQL query is generated based on the question you have provided. After creating a query with the tool you can add the generated SQL to your querybook with a single click.

Once you have created a query, the query can be saved for later use by yourself or others with access to the project. Query results can be exported in CSV or JSON formats. You can create visualizations directly in the Query editor.

Review query history

The query history tool displays past queries that you have run. To navigate to the Query history page, choose the **Query history** icon from the side navigation of the query editor. Select a data source to view query results from that source.

Queries are sorted so that the most recent queries are listed first. Each query contains the following information:

- **Execution ID.** This is a unique ID for the query that you ran at a specific time.
- **Status.** This displays the status of a query. The following status types can be displayed:
 - **Completed.** This means the query was successfully executed and has completed running. You can view results for it.
 - **Failed.** A query can fail for a number of reasons. To see more details about the query and troubleshoot it, choose the execution ID of the query you want to see details for.
 - **Canceled.** A query is canceled if you choose the Stop button on the Query editor page before the query finishes running.
 - **Running.** The query editor might show this status if you are querying a large dataset. After some time, this status will be updated when the query is complete.
- **Duration.** This shows how long it took for the query to finish running.
- **SQL.** This shows a preview of the SQL you used to run the query. To view the complete SQL, choose the execution ID of a query to see details for it.
- **Start time.** This displays the time and date that the query was started. Queries outside of the specified time range at the top of the page are not displayed.
- **Engine.** This specifies the engine that was used to perform the query.

Filtering the query history

You can filter query history results by data source, status, and time range. By default, the query history page displays queries of all status types from the past 24 hours.

To display queries from a different data source, choose Data source and select the data source you want to see queries from.

To display queries with a certain status, such as Cancelled or Running, choose Status and then choose the type of status that you want to display. You can also choose All to view queries of all status types.

To view queries from a different time frame, complete the following steps:

1. Choose **Time range**.
2. Select the time range you want to see queries from.
3. (Optional) If you choose a custom date range, select the time zone you want it in.
4. Choose **Apply**.

Reviewing additional details

To see more details about a query you have run, select the execution ID of the query you want to see details for. A side window then opens and displays additional details about the query, including a display of the SQL that was used to run the query. The details displayed are different depending on the data source of the query. For instance, in a query for Amazon Athena you can view AWS S3 encryption information and query stats that show a visual representation of the time it took to run the query.

To view the full SQL and results of the query, choose **Open in a new tab**. This opens the query in the query editor tool.

Schedule a query

There are two ways to schedule queries in Amazon SageMaker Unified Studio.

- You can schedule your queries directly in the editor. This way you can schedule a single query quickly. The following sections contain information about this method.
- You can schedule your queries using a DAG and the workflows interface. This way you can combine multiple elements in the same schedule. For more information about this method, see [Workflows](#).

Create a schedule for a query in Amazon SageMaker Unified Studio

You can create a schedule directly in the query editor interface in Amazon SageMaker Unified Studio. To do this, use a project with the **All capabilities** project profile or another project profile

with scheduling enabled in the Tooling blueprint parameters. If you have created a project that needs to be updated to enable scheduling, contact your admin.

To create a schedule in the query editor, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the query editor. You can do this by choosing **Query Editor** from the **Build** menu of a project.
3. After creating and saving your query, choose **Actions**, then **Create schedule**.

 **Note**

You must have a connection selected in order to create a schedule.

4. Under **Schedule name**, enter a name for the schedule.
5. Under **Schedule status**, choose an option to determine whether the schedule will begin running after being created.
 - Choose **Active** to activate the schedule and run the Visual ETL flow when the schedule indicates it should run.
 - Choose **Paused** to create a schedule that will not run the Visual ETL flow yet.
6. (Optional) Write a description of the schedule.
7. Choose a schedule type.
 - Choose **One-time** to run the Visual ETL flow at one specific time.
 - Choose **Recurring** to create a schedule that runs the Visual ETL flow at multiple times that you choose.
8. Choose the days and times that the schedule will run.
9. Choose **Create schedule**.

Review scheduled queries in Amazon SageMaker Unified Studio

You can review scheduled queries in the query editor in Amazon SageMaker Unified Studio. On the schedules page of the query editor, you can pause, edit, and delete schedules. You can also view

the status and other information for a schedule and choose the name of a schedule to view runs and additional data.

To review scheduled queries, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to your project.
3. Choose **Query Editor** from the **Build** menu.
4. Choose the Scheduled queries icon in the left navigation of the query editor.

You can then pause, edit, or delete a schedule by choosing the three-dot **Actions** menu next to a schedule in the list.

To view information about different times the schedule has run, choose the name of the schedule to view the **Runs** section for that schedule. You can choose the name of a run to see querybook output and a log for that run.

Using workflows in Amazon SageMaker Unified Studio

With a Amazon SageMaker Unified Studio workflow, you can set up and run a series of tasks in Amazon SageMaker Unified Studio. Amazon SageMaker Unified Studio workflows use Apache Airflow to model data processing procedures and orchestrate your Amazon SageMaker Unified Studio code artifacts. Go to the **Workflows** page of a project in Amazon SageMaker Unified Studio to create workflows in Python code, run them, and review logs.

Note

Workflows are available in Amazon SageMaker Unified Studio projects created with the **All capabilities** project profile.

To use workflows in Amazon SageMaker Unified Studio, you must provision an instance of at least 4GB memory and 4vCPUs.

There are two compute spaces that you can use for workflows in your project:

- **Local space.** Only you can view and edit the workflows in your local space.
- **Shared environment.** Everyone in the project sees and accesses the files in the shared environment.

Each workflow that you create starts in your local space as a file. To share your workflows with other users, commit the file defining your workflow and sync the workflow with a shared environment.

Create a workflow in Amazon SageMaker Unified Studio

Use workflows to orchestrate notebooks, querybooks, and more in your project repositories. With workflows, you can define a collection of tasks organized as a directed acyclic graph (DAG) that can run on a user-defined schedule.

Prerequisites

Before you can create a workflow, you must prepare the files that you want to run. The files should be saved in your JupyterLab space in a folder that you can easily locate later. You must

also provision an instance of at least 4GiB memory and 4vCPU in a project created with the **All capabilities** project profile.

To provision an instance for workflows

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that you want to create your workflow in.
3. Expand the **Build** menu in the top navigation, then choose **JupyterLab** to navigate to the JupyterLab IDE.
4. Choose **Configure space**
5. Under **Instance**, select an instance that has at least 4 vCPU and 4GiB. This might result in an additional cost.
6. Choose **Save and restart**. It might take a few minutes for the restart to finish.

If you want to schedule a query to run, you must first save the querybook to the project and pull it into your JupyterLab space. The steps are as follows:

To prepare to schedule a query

1. In a project that uses the **All capabilities** project profile, create the query you want to run and save it to the project. For more information, see [the section called “Create a query”](#).
2. Expand the **Build** menu in the top navigation, then choose **JupyterLab** to navigate to the JupyterLab IDE.
3. Choose the **Git** icon in the left navigation.
4. Choose the **Pull latest changes** icon to do a git pull and bring the published querybook into your JupyterLab space.
5. Note the location of the file in the JupyterLab file navigation. You will need that path later so you can add it to your workflow.

Create a workflow

To create a workflow, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

2. Navigate to a project that was created with the **All capabilities** project profile. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. In the **Build** menu, choose **Workflows**. This takes you to the Workflows page.
4. Choose **Create workflow in editor**. This takes you to the Code page and opens a new notebook file in the `workflows/dags` folder of the JupyterLab file navigation. The file is prepopulated with a workflow definition template.
5. Update the file as desired to create your workflow.
 - a. Update `WORKFLOW_SCHEDULE` to determine when the workflow will be scheduled to run.
 - b. Update `NOTEBOOK_PATH` to point to the querybook or JupyterLab notebook that you want to run. For example, '`src/querybook.sqlnb`'.
 - c. Update `dag_id` with an ID that you can identify later.
 - d. Add tags and parameters, if desired. For more information, see [Params](#) in the Apache Airflow documentation.

When you create a workflow, you are modifying the directed acyclic graph (DAG) within the Python file. A DAG defines a collection of tasks with their dependencies and relationships to show how they should run.

A DAG consists of the following:

- A [DAG](#) definition. The DAG ID will also be the name of the workflow.
- [Operators](#) that describe how to run the DAG and the [tasks](#) to run.
- [Operator relationships](#) that describe the order in which to run the tasks.

For more information about DAGs, see [DAGs](#) in the Apache Airflow documentation. You can configure a DAG to run on a schedule or run it manually.

You can include multiple DAGs to create multiple workflows. When you have included the DAGs you want to use, save the file in the `workflows/dag` folder in JupyterLab. There might be a slight delay before the workflow appears on the Workflows page.

Sample workflow

A sample workflow definition can be seen here.

```
from airflow.decorators import dag
from airflow.utils.dates import days_ago
from workflows.airflow.providers.amazon.aws.operators.sagemaker_workflows \
    import NotebookOperator

#####
#
# Enter in your desired schedule as WORKFLOW_SCHEDULE. Some options include:
#
# '@daily' (daily at midnight)
# '@hourly' (every hour, at the top of the hour)
# '30 */3 * * *' (a CRON string, run at minute 30 past every 3rd hour)
# '0 8 * * 1-5' (a CRON string, run every weekday at 8am)
#
#####

WORKFLOW_SCHEDULE = '@monthly'

#####
#
# Enter in the path to your artifacts. Example:
# 'src/example_notebook.ipynb'
#
#####

PROCESS_PATH = 'src/dataflows/airQualityToLakehouse.vetl'
QUERY_PATH = 'src/QueryBrooklynDataPutInS3.sqlnb'

default_args = {
    'owner': 'alexa',
}

@dag(
    dag_id='air-quality-process-and-query',
    default_args=default_args,
    schedule_interval=WORKFLOW_SCHEDULE,
    start_date=days_ago(2),
    is_paused_upon_creation=False,
    tags=['example-project', 'alexa'],
    catchup=False
)
def air_quality():
```

```
def process_data():
    return NotebookOperator(
        task_id="process-data",
        input_config={'input_path': PROCESS_PATH, 'input_params': {}},
        output_config={'output_formats': ['NOTEBOOK']},
        wait_for_completion=True,
        poll_interval=5
    )

def query_data():
    return NotebookOperator(
        task_id="query-data",
        input_config={'input_path': QUERY_PATH, 'input_params': {}},
        output_config={'output_formats': ['NOTEBOOK']},
        wait_for_completion=True,
        poll_interval=5
    )

process_data() >> query_data()

air_quality = air_quality()
```

View workflow details

After you create a workflow, it appears in a list on the [Workflows](#) page in Amazon SageMaker Unified Studio. On the Workflows page, you can see each workflow you created with the name you defined using the DAG ID.

To view details about workflow runs and parameters, select the name of a workflow from the list on the [Workflows](#) page in Amazon SageMaker Unified Studio.

- On the **Runs** tab, you can view the results of running the workflow. You can filter to show successful runs. This page shows information about the workflow run triggers, durations, and timeframes. There is also an **Actions** column where you can choose to stop a workflow if it is still running. There is a limit of 1000 rows on the **Runs** tab for a workflow.

To view more details about a run, choose the name of a run. This takes you to the run details page, with information about the tasks and parameters in the workflow. You can view which tasks were successfully completed on the **Task log** tab.

To view more details about a run, choose the name of a run. This takes you to the run details page, with information about the tasks and parameters in the workflow. You can view which tasks were successfully completed. For workflows that run Python notebooks and not querybooks, you can view the output in the **Notebook output** tab. This can be useful for viewing tasks in more detail and troubleshooting if needed.

- The **Default parameters** tab shows the default parameters outlined in the workflow code. To modify the parameters, choose **Edit code** and edit parameters. For more information about parameters, see [Params](#) in the Apache Airflow documentation.
- The **Definition** tab shows the code used for the workflow. This matches the code you wrote on the Code page. Choose **Edit code** to navigate back to the Code page and make changes.
- The **Tags** tab shows optional tags that are defined in the workflow definition file. These are Airflow tags, not AWS tags. For more information, see [Add tags to DAGs and use it for filtering in the UI](#) in the Apache Airflow documentation.

Run a workflow

To run a workflow, navigate to the workflow details page by selecting a workflow from the Workflows page list. Then choose Run. You can then choose one of the following two options:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to a project that was created with the **All capabilities** project profile. To do this, use the center menu at the top of the landing page and choose **Browse all projects**, then choose the name of the project that you want to navigate to.
3. In the **Build** menu, choose **Workflows**. This takes you to the Workflows page.
4. Choose the name of a workflow to navigate to the workflow details page.
5. Expand the **Run** menu, then choose one of the following options:
 - Run with default parameters. This option starts running the workflow using the parameters already defined in the DAG file. To review these parameters, see the **Default parameters** tab.
 - Run with custom parameters. This option opens a window where you can change the inputs for the parameters defined in the DAG file. Enter the variables you want to use, and then choose **Start run** to start running the workflow.

The workflow run then appears on the **Runs** tab of the workflow details page. The workflow runs until it is complete or until you choose to stop it.

Running a workflow puts tasks together to orchestrate Amazon SageMaker Unified Studio artifacts. You can view multiple runs for a workflow by navigating to the Workflows page and choosing the name of a workflow from the workflows list table.

If you want to see more runs, you can view them using the Airflow UI. Navigate to the Workflows page, choose the three dots in the Action column for a workflow, then choose **Open Airflow UI**. This page displays charts and graphics about the workflow.

 **Note**

To open the Airflow UI, your browser should allow cross-site cookie sharing. If you receive an error message, check the cookie settings in your browser.

Share a workflow with other project members in an Amazon SageMaker Unified Studio workflow environment

After a workflow environment has been created by a project owner, any project member can sync their files to share them in the environment. After you sync your files, all project members can view the workflows you have added in the workflow environment. Files that are not synced can only be viewed by the project member that created them.

To share your workflows with other project members a workflow environment, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to a project that was created with the **Data analytics and AI-ML model development** project profile. You can do this by using the center menu at the top of the page and choosing **Browse all projects**, then choosing the name of the project that you want to navigate to.
3. In the **Build** menu, choose **JupyterLab**.
4. Locate the workflow you want to share in the workflows/dags folder.
5. Choose the **Git** icon in the left navigation.
6. Choose the **+** icon next to the files you want to commit.

7. Enter a brief summary of the commit in the **Summary** text entry field.
8. (Optional) Enter a longer description of the commit in the **Description** text entry field.
9. Choose **Commit**.
10. Choose the **Push committed changes** icon to do a git push.
11. In the **Build** menu, choose **Workflows**. This takes you to the Workflows page.
12. On the **Shared environment** tab, choose **Sync files from project**.
13. Choose **Confirm**.

Workflow environments in Amazon SageMaker Unified Studio

Use a shared workflow environment to share workflows with other project members. Workflow environments must be created by project owners. To update or delete a workflow environment, you must be an owner of the project that the workflow environment is in. After a workflow environment has been created by a project owner, any project member can sync their files to share them in the environment.

Only one workflow environment can exist in a project at a time.

Create a workflow environment

To create a workflow environment, you must be an owner of the project that you want to create a workflow environment in.

To create a workflow environment, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to a project that was created with the **Data analytics and AI-ML model development** project profile. To do this, use the center menu at the top of the landing page and choose **Browse all projects**, then choose the name of the project that you want to navigate to.
3. In the center menu, choose **Compute**. This takes you to the Compute page.
4. On the **Workflow environments** tab, confirm that there are no workflow environments in the project yet. Then choose **Create**.
5. In the **Create workflow environment** window, review the parameters of the workflow environment. These are determined by your admin. If you want any of these parameters to change, contact your admin.

6. Choose **Create workflow environment**.

 **Note**

Workflow environment creation takes several minutes to complete.

Update a workflow environment

To update a workflow environment, you must be an owner of the project that you want to update a workflow environment in.

To update a workflow environment, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.
2. Navigate to the project that contains the workflow environment that you want to update. To do this, use the center menu at the top of the landing page and choose **Browse all projects**, then choose the name of the project that you want to navigate to.
3. In the center menu, choose **Compute**. This takes you to the Compute page.
4. On the **Workflow environments** tab, expand the **Actions** menu and choose **Update**.
5. Choose **Update workflow environment**.

 **Note**

Updating a workflow environment takes several minutes to complete.

Delete a workflow environment

To delete a workflow environment, you must be an owner of the project that contains the workflow environment that you want to delete.

To delete a workflow environment, complete the following steps:

1. Navigate to Amazon SageMaker Unified Studio using the URL from your admin and log in using your SSO or AWS credentials.

2. Navigate to the project that contains the workflow environment that you want to delete. To do this, use the center menu at the top of the landing page and choose **Browse all projects**, then choose the name of the project that you want to navigate to.
3. In the center menu, choose **Compute**. This takes you to the Compute page.
4. On the **Workflow environments** tab, expand the **Actions** menu and choose **Delete**.
5. Confirm the action by typing `confirm`, then choose **Delete workflow environment**.

 **Note**

Deleting a workflow environment takes several minutes to complete.

Using the Amazon SageMaker Unified Studio chat assistant

Amazon SageMaker Unified Studio includes an integrated Amazon Q chat assistant to help you get answers and assistance while using the platform. The chat assistant uses natural language processing to understand your questions and provide relevant responses using the Amazon SageMaker Unified Studio documentation and AWS knowledge base.

With the integrated chat assistant, you can do the following:

- Ask questions about using Amazon SageMaker Unified Studio features, troubleshooting issues, AWS services, and more using natural language.
- Get prompt suggestions for common queries.

To get started, choose the Amazon Q icon in the top right corner of a project in Amazon SageMaker Unified Studio.



Then type your question into the chat input box or choose from the prompted suggestions. The assistant analyzes your input and provides a tailored response or guide you through next steps. You can continue the dialogue across multiple turns, and the chat history is maintained within your session.

The Amazon Q chat assistant provides a readily accessible source of information and troubleshooting assistance within the development environment.

The screenshot shows the Amazon Q interface. At the top, there is a logo with a hexagon containing a circle and the text "Amazon Q". To the right are settings and close buttons. Below the header, a main message reads: "I am Amazon Q, your generative AI assistant. Ask me anything about Amazon SageMaker Unified Studio and other AWS services. Or, choose a sample question below to start a conversation." Below this message are four sample questions in rounded rectangular boxes: "What is SageMaker Unified Studio?", "Find me data on marketing", "How do I connect to a Redshift cluster?", and "How can I build a Gen AI playground in Bedrock?". At the bottom, there is a text input field with the placeholder "Ask me anything about SageMaker Unified Studio" and a send button with a right-pointing arrow. Below the input field, it says "Max 1000 characters". A note at the bottom states: "Amazon Q Developer uses generative AI. You may need to verify responses. See the [AWS Responsible AI Policy](#)".

I am Amazon Q, your generative AI assistant.

Ask me anything about Amazon SageMaker Unified Studio and other AWS services. Or, choose a sample question below to start a conversation.

What is SageMaker Unified Studio?

Find me data on marketing

How do I connect to a Redshift cluster?

How can I build a Gen AI playground in Bedrock?

Ask me anything about SageMaker Unified Studio ➤

Max 1000 characters

Amazon Q Developer uses generative AI. You may need to verify responses. See the [AWS Responsible AI Policy](#)

Security in Amazon SageMaker Unified Studio

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from data centers and network architectures that are built to meet the requirements of the most security-sensitive organizations.

You can see the [Security](#) section in the Amazon SageMaker Unified Studio Administrator Guide for detailed information and instructions on how to apply the shared responsibility model when using Amazon SageMaker Unified Studio. The topics in the [Security](#) section in the Amazon SageMaker Unified Studio Administrator Guide show you how to configure Amazon SageMaker Unified Studio to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Amazon SageMaker Unified Studio resources.

Security is a shared responsibility between AWS and you. The [shared responsibility model](#) describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the [AWS Compliance Programs](#). To learn about the compliance programs that apply to Amazon SageMaker Unified Studio, see [AWS Services in Scope by Compliance Program](#).
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

Quotas for Amazon SageMaker Unified Studio

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is account-specific and Region-specific.

The resource quota limits are applied at the account level, meaning the depletion of resource quotas in one project can affect all other projects within the account. Administrators can monitor resource usage by project and take necessary actions through the

Amazon SageMaker Unified Studio has the following quotas and limits.

Resource	Default
Maximum number of JupyterLab instances	2500
Maximum number of project members for your Amazon SageMaker platform domain. The total number of project members is the product of project members and projects.	2500
Maximum number of spaces	2500
Maximum number of projects	2500
Maximum number of Micro environments	200

For more information about other AWS service quotas, see [AWS service quotas](#).

For more quotas information, see the following:

- [Amazon SageMaker Supported Regions and Quotas](#)
- [Amazon Managed Workflows for Apache Airflow endpoints and quotas](#)
- [Amazon Redshift endpoints and quotas](#)
- [Amazon EMR endpoints and quotas](#)
- [Amazon DataZone endpoints and quotas](#)
- [Amazon Q Business endpoints and quotas](#)
- [Amazon Athena endpoints and quotas](#)

- [Amazon Bedrock endpoints and quotas](#)
- [AWS Glue endpoints and quotas](#)

Troubleshooting in Amazon SageMaker Unified Studio

Troubleshooting AWS Lake Formation permissions for Amazon SageMaker Unified Studio

This section contains troubleshooting instructions for issues that you might encounter when you [Configure Lake Formation permissions for Amazon SageMaker Unified Studio](#).

Error message	Resolution
Unable to assume the IAM role.	<p>This error is thrown when Amazon SageMaker Unified Studio is not able to assume the IAM role needed for updating AWS Lake Formation permissions. To fix the issue, go to the IAM console in the account where your data asset exists and make sure that the the IAM role you see in the error message has a trust relationship with the Amazon DataZone service principal.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"><pre>{ "Effect": "Allow", "Principal": { "Service": "datazone .amazonaws.com" }, "Action": ["sts:AssumeRole", "sts:SetContext"], "Condition": { "StringEquals": { "aws:SourceAccount ": <i>accountID</i> } } }</pre></div>

Error message	Resolution
The IAM role <i>role-arn</i> does not have the necessary permissions to read the metadata of the asset you are trying to subscribe to.	This error is thrown when Amazon SageMaker Unified Studio is able to assume the IAM role but the role does not have the necessary permissions. To fix the issue, go to the IAM console in the account where your data asset exists and make sure that the role you see in the error message has the <u>SageMaker StudioProjectUserRolePolicy</u> attached to it.
Asset is a resource link. Amazon SageMaker Unified Studio does not support subscriptions to resource links.	This error is thrown when the asset you are trying to publish to Amazon SageMaker Unified Studio is a resource link to an AWS Glue table.
Asset is not managed by AWS Lake Formation.	This error means that the AWS Lake Formation permissions are not enforced on the asset you are trying to publish. This can happen because the Amazon S3 location of the asset is not registered in AWS Lake Formation. To fix the issue, log into the AWS Lake Formation console in the account where the table exists and register the Amazon S3 location in AWS Lake Formation mode or hybrid mode.

Error message	Resolution
IAM role <i>role-name</i> does not have necessary AWS Lake Formation permissions to grant access to this asset.	<p>This error means that the IAM role in the error message does not have the necessary permissions for Amazon SageMaker Unified Studio to be able to manage permissions on the published table. You can resolve the issue by granting the following AWS Lake Formation permissions to the the IAM role on the table you are looking to publish.</p> <ol style="list-style-type: none">1. Describe and Describe Grantable permissions on the database where the tables exist2. Describe, Select, Describe Grantable , and Select Grantable permissions on the all the tables in the above database that you want Amazon SageMaker Unified Studio to manage access for on your behalf.

Amazon EBS Volume Depletion with Local Notebook Execution

Question

"Enhance the StartExecution API response when throttling occurs due to low disk space, instructing the user to delete files from the jobs folder."

Answer

1. Navigate to JupyterLab
2. In the Jobs folder, select the folder and files
3. Select delete

Domain

Question

From IAM SSO access portal URL, SAmazon SageMaker Unified Studio is not listed. When I click on Amazon DataZone, Amazon DataZone portal is shown. I clicked on SIGN IN WITH SSO, it failed due to Invalid redirectUri provided.

Answer

Visit DAmazon DataZone console, choose your domain, then click the Amazon SageMaker Unified Studio URL.

SAML Identity Provider Email Issue

Question

When using 3rd party SAML identity providers, the domain creation flow does not identify my email address.

Answer

This happens because during the user provisioning step, the email field was not populated in your local SSO instance. When sync-ing with 3rd party SAML identity providers, modify the default mapping to ensure it includes the "email" field and re-do the sync.

Project Creation Failure

Question

When I configure a project profile with resources pointing to another region/account, and try to create a new project using the project profile, it failed due to the error Project creation failed because one or more resources could not be provisioned.

Answer

Make sure that you complete following configurations:

1. Domain owner account: In the Domains menu, choose your domain. Under the Account associations tab, verify that domain is associated with the target account, and the status is Associated.

2. Target account: In the Associated domains menu, choose the associated domain. Choose your blueprint. Under the Regions tab, verify that the target region is added. Under the Authorization tab, verify that the target domain unit is shown.
3. Domain owner account: In the Domain details, under the Project profiles tab, choose your project profile. Under the Blueprint deployment settings tab, choose Name of your blueprint, under Deployment order, verify that Account ID and Region are configured correctly. Under the Authorized users and groups, verify that your SSO user is added.

Data Explorer Visibility Issue

Question

On the data explorer, I cannot see my existing databases and tables on Glue Data Catalog. How can I query them?

Answer

Amazon SageMaker Unified Studio configures AWS IAM permissions and permission boundaries. You can optionally remove the permission boundaries to allow access to the existing databases and tables.

Data Catalog Visibility Issue

Question

On the data catalog, I cannot see my existing databases and tables on Glue Data Catalog. How can I view them?

Answer

1. On your project page, choose Data sources.
2. Choose CREATE DATA SOURCE to add the existing databases and tables as a data source.
3. For Data source type, choose AWS Glue, and choose NEXT.
4. Configure how to select your databases and tables here.
5. Once everything is filled, choose NEXT and go ahead to register the data sources.

Connection to Amazon RDS MySQL in Existing VPC

Question

I want to connect to my Amazon RDS MySQL database instance that exists in my existing VPC. When I add a connection, I do not see any settings about VPC. How can I configure the reachability?

Answer

Amazon SageMaker Unified Studio uses the VPC and subnets that are specified in the domain creation. If you have the data source in a separate VPC, you can configure network reachability between the domain VPC and the data source VPC using VPC peering or Transit Gateway, or alternatively you can create a new domain using the data source VPC.

Visual ETL Flow Column Selection

Question

I created a data source, and now I am adding a new transform on top of it. But I cannot choose the columns for the transform.

Answer

When you start authoring a visual ETL flow, the data preview is also started. Once the preview is completed, then schema is automatically collected and available for further transforms.

Invalid or expired auth token when accessing an IDE

Question

When I tried to access an IDE in Amazon SageMaker Unified Studio I got an invalid or expired auth token error message. How do I resolve this?

Answer

This issue often occurs when third-party cookies are blocked. Amazon SageMaker Unified Studio uses the third-party cookies for authenticating into an Amazon SageMaker Unified Studio IDE since the IDE is hosted in a separate domain from the primary site. To help resolve this issue, ensure that third-party cookies are allowed in your browser.

Note

To do this in Safari, go to **Preferences** or **Settings** and ensure that on the **Privacy** page, the box next to **Prevent cross-site tracking** is unchecked.

For more information, see the following sites:

- For Firefox, see [Third-party cookies and Firefox tracking protection](#).
- For Safari, see [Prevent cross-site tracking in Safari on Mac](#).
- For Chrome, see [Delete, allow and manage cookies in Chrome](#).
- For Edge, see [Manage cookies in Microsoft Edge: View, allow, block, delete and use](#).

JupyterLab Configure Magic Error

Question

When I ran %%configure magic, it returned the error Connection name cannot be empty.

Answer

The magic syntax is different from Glue Interactive Session's existing kernel. Instead, run the magic with following syntax:

```
%%configure --name (compute) (-f)
{
  "key": "value"
}
```

For example, if you want to change the default Spark SQL catalog name for project default Spark connection, run following magic:

```
%%configure --name project.spark --f
{
  "--conf": "spark.sql.defaultCatalog=glue_catalog"
}
```

Document history for the Amazon SageMaker Unified Studio User Guide

The following table describes the documentation releases for Amazon SageMaker Unified Studio.

Change	Description	Date
<u>New feature</u>	You can now use a chat agent app as a node in an Amazon Bedrock in SageMaker Unified Studio flow app	May 6, 2025
<u>General release</u>	Amazon SageMaker Unified Studio is now generally available. This also includes updates to the full capability project profile, additional data connections, new compute resources, and more.	March 13, 2025
<u>Initial release</u>	Initial release of the Amazon SageMaker Unified Studio User Guide	December 3, 2024