onestream

SensibleAI Studio Guide

PV9.1.0 SV120

Table of Contents

Overview

This document details the SensibleAI Studio user interface, including functionality and requirements of each page. Information includes:

- How to interact with each section and page in the solution.
- How to navigate all routines in the Routine Library and create a Routine Instance.
- How to invoke and view a Routine Method Run.

Explore

The first major section of SensibleAI Studio is the Explore page. On this page, users can select from a list of all the available Routines and choose which to use for creating a new Routine Instance.

Click on Routine Name to see more information on the right side of the page; pages include Definition, Signature, and Metadata.

- **Definition**: This section displays details about the Routine, its use cases, and the creation date.
- **Signature**: This section displays details about methods, required configuration inputs, outputs, and default memory capacity.
- **Developer Docs**: This section provides creators and power users with the necessary information to invoke the selected routine programmatically. This information includes raw method names and their corresponding artifact qualified keys, if applicable.

Routine Library: This is where the individual Routines are stored within the solution. The total number of Routines here increases as future iterations of SensibleAI Studio are developed and released.

Run

The **Run** page is the next major section of SensibleAI Studio. Here, users can create new Routine Instances and Routine Runs or interact with existing ones. Users can also click the self-icon in each section to display only their personal instances or runs.

Selecting a Routine Instance will automatically filter the Routine Runs Activity grid to show only the runs associated with that instance. Additional filters for Routine Runs such as **Stateful**, **Stateless**, **Today**, **Completed**, **Running**, and **Queued** can be applied individually or in combination to further customize the view.

By selecting a Routine Instance or Routine Run, users can view its **status**, **summary**, **artifacts**, and **documentation** on the right side of the page. Depending on the selection, users also have access to a range of available actions tailored to either the instance or the run.

For more information, see the Routine Instances and Routine Runs sections.

Setup and Installation

This section contains details for planning, configuring, and installing the SensibleAI Studio solution. Before you install the solution, familiarize yourself with these details.

Solution Info

Solution Version:

STU-SV120-XPFv4.1.0-PV910

Xperiflow Version:

4.1.0

Minimum Routine Memory Capacity:

1 GB

Maximum Concurrent Routine Runs:

10

Total Environment Memory Capacity:

10 GB

See OneStream Solution Modification Considerations.

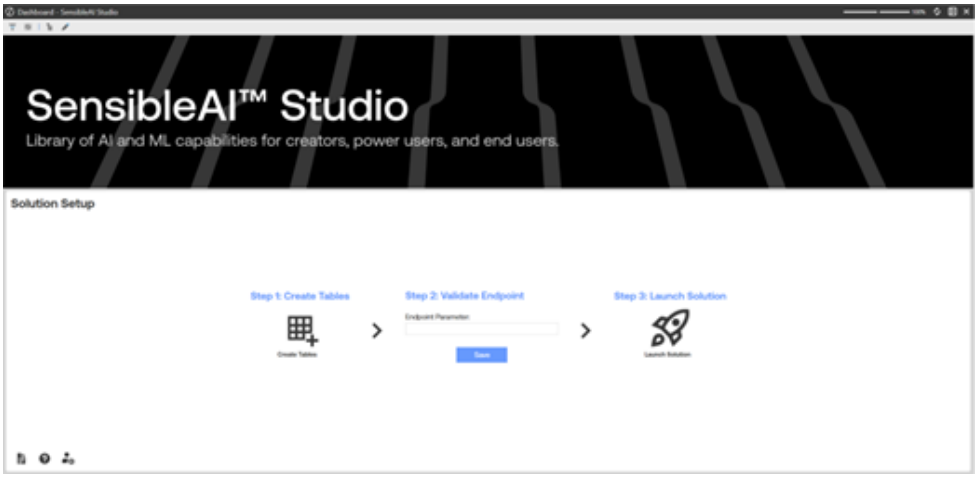
Dependencies

Component	Description
OneStream 9.1.0 or later	Minimum OneStream Platform version required to install this version of SensibleAI Studio.
Xperiflow 4.1.0 or later	Minimum version required to install this version of SensibleAI Studio.
Xperiflow Business Rules V210 (XBR)	External API client library to allow STU to interface with the Xperiflow Engine. The required version of XBR is packaged with all STU versions.

Set Up SensibleAI Studio

Follow these steps to set up SensibleAI Studio:

1. Download the SensibleAI Studio Solution from the OneStream Solution Exchange.
2. After the OneStream support team ensures that the proper contract is in place, a link is sent to download the SensibleAI Studio solution. To complete the setup, the user must now set the proper endpoint parameter and launch the solution.
3. Follow the outlined Solution Setup steps:

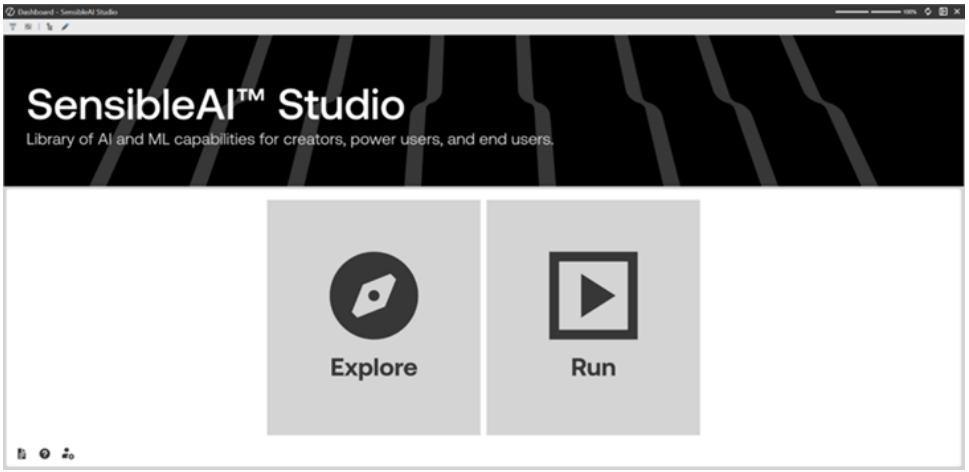


a. Create Tables

b. Validate Endpoint (will vary based on the environment)

c. Launch Solution

4. When a user reaches the Home page after clicking Launch, SensibleAI Studio has been successfully set up correctly:



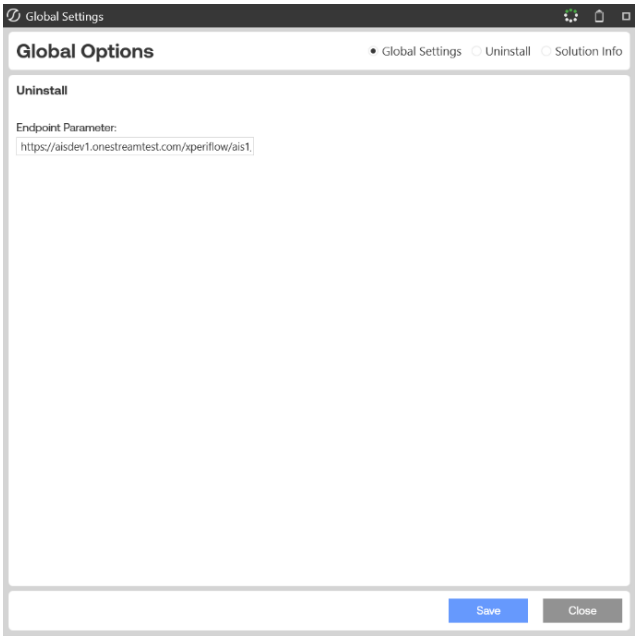
Settings

To access the Global Options page, click **Settings**( ) in the bottom left corner of the **Explore** page or the **Run** page.

Global options include:

- Global Settings
- Uninstall
- Solution Info

Global Settings

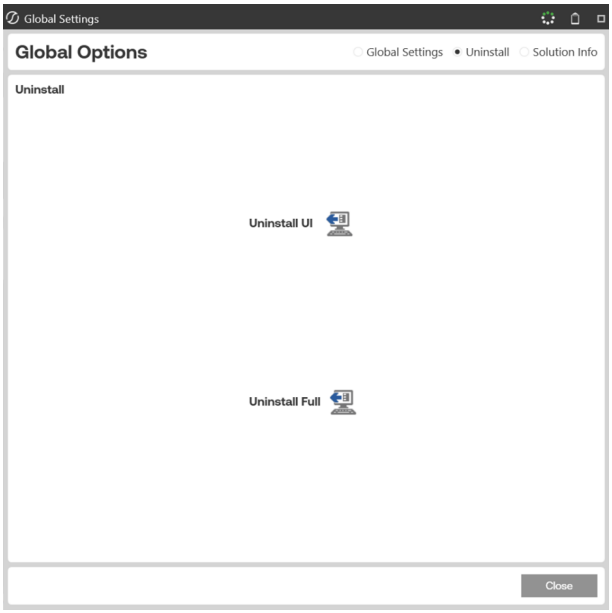


Endpoint Parameter:

Predefined endpoint to access the application.

> **CAUTION:** Do not make changes to this value unless instructed to do so.

Uninstall



There are two uninstall options:

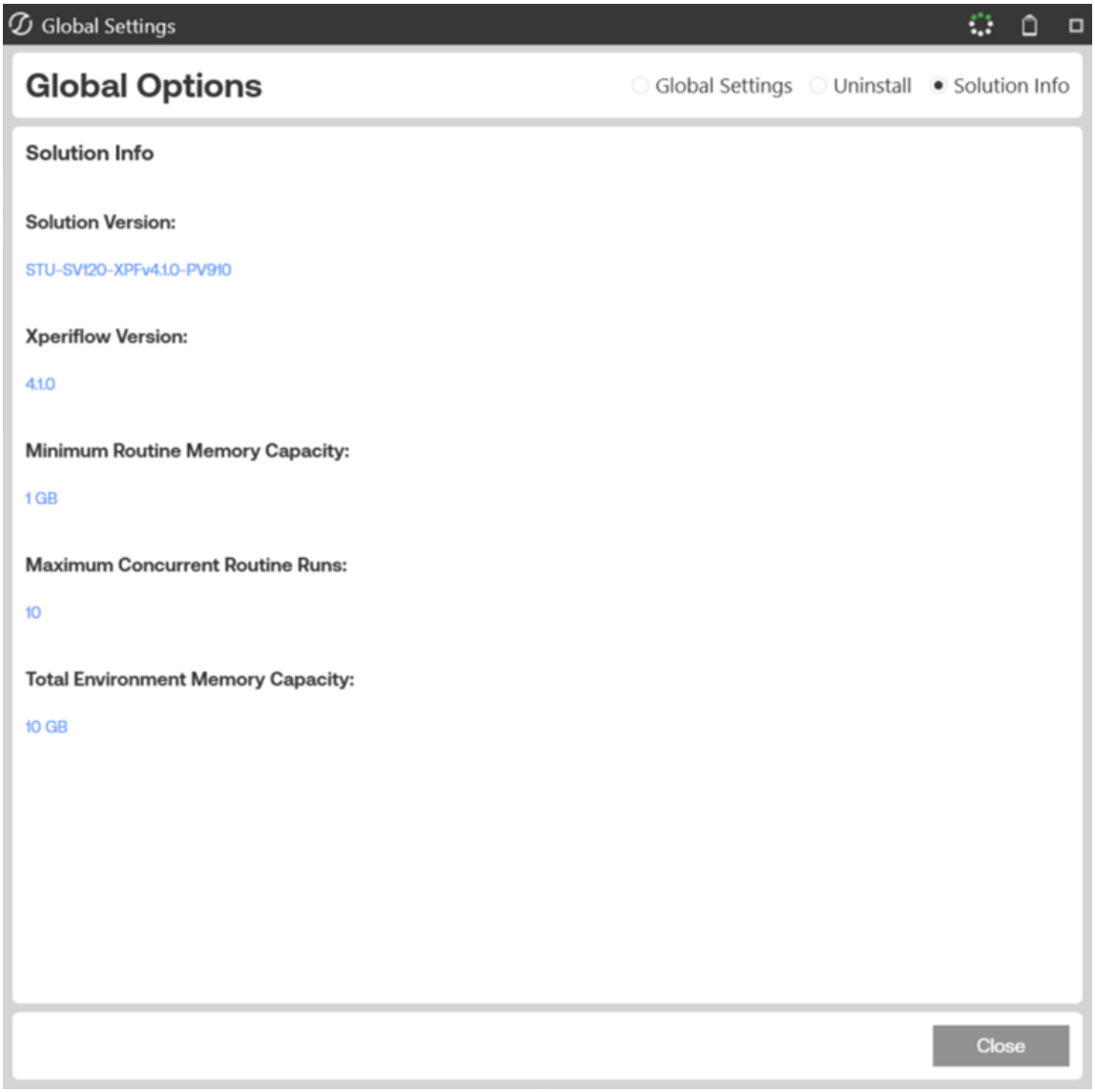
> **CAUTION:** Uninstall procedures are irreversible.

Uninstall UI removes SensibleAI Studio, including related dashboards and business rules, but leaves the database and related tables in place. Choose this option if you want to accept a SensibleAI Studio update without removing data tables.

Uninstall Full removes all related data tables, data, SensibleAI Studio dashboards, and business rules. Choose this option to completely remove SensibleAI Studio or to perform an upgrade that is so significant in its changes to the data tables that this method is required.

Solution Info

The Solution Version is comprised of the Solution Code, Solution Version, Xperiflow Version, and OneStream platform version (Solution Code-Solution Version-Xperiflow Version-OneStream Platform Version).

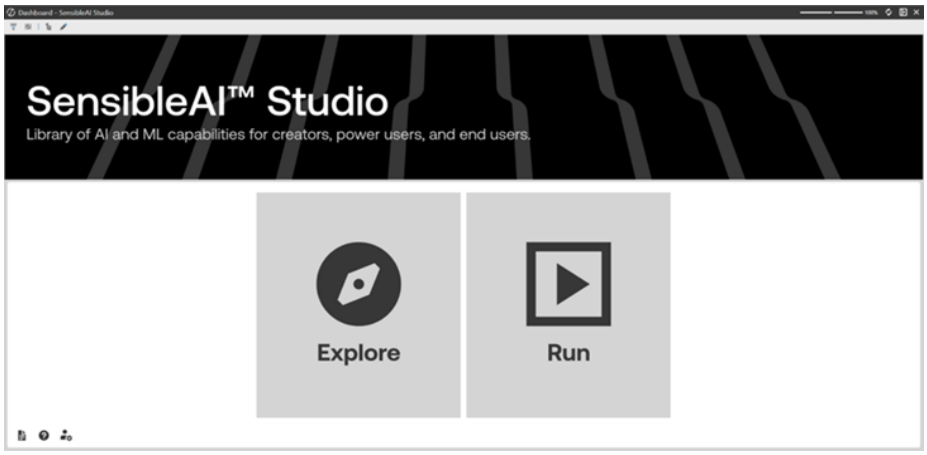

Global Settings
Global Options
Global Settings
Uninstall
Solution Info
Solution Info
Solution Version:
STU-SVt20-XPFv4.1.0-PV910
Xperiflow Version:
4.1.0
Minimum Routine Memory Capacity:
1 GB
Maximum Concurrent Routine Runs:
10
Total Environment Memory Capacity:
10 GB
Close

Navigate in SensibleAI Studio

The following sections describe the ways to navigate the SensibleAI Studio solution.

SensibleAI Studio Home Page

The Home page displays a clickable interface prompting the user to launch either the Explore page or the Run page.



Once launched, the **Refresh** on the top right of the screen allows users to refresh the current page. The **Home** button to the right brings users back to the Home page.

SensibleAI Studio Sections

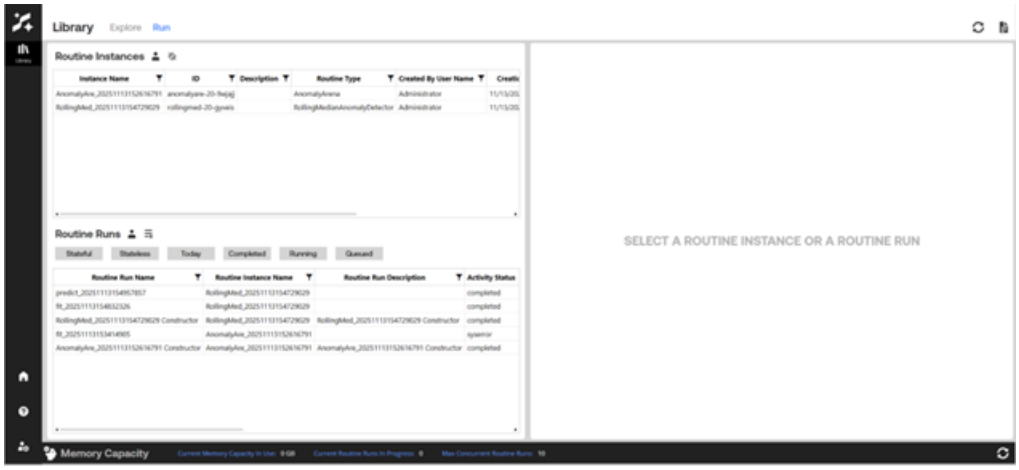
The left navigation panel includes a single section, **Library**. The top left corner of the upper toolbar shows the pages available in the selected section. Below are the available pages and their components:

Explore Page

Within the Routine Library, click a Routine Name to view documentation for all versions of that Routine. Documentation includes a Definition section, Signature information, and Developer Docs. Users can create a Routine Instance or access the Routine Settings configurations.

Run Page

Interact with existing Routine Instances and Routine Runs, or create new ones as needed. Whether or not an instance or run is selected, users can access additional details in the **Summary** section, view method-run outputs in the **Artifacts** section, and explore routine-specific information in the **Documentation** section. For more information, see Routine Instances and Routine Runs sections.



Toolbar Icons

Icon	Description
Home	Navigates to the Home page.
Help	Opens the SensibleAI Studio User Guide.
Settings	Opens the SensibleAI Studio settings. Configure Global Settings options and uninstall the solution.
Refresh	Refreshes the current page.

Icon	Description
Activity Log	Displays all job activity from the AI Services engine for jobs executed within SensibleAI Studio.

Each section page includes **Refresh** and **Activity Log** buttons at the top right and **Home**, **Help**, and **Settings** buttons at the lower left.

Routines

A Routine is an AI or ML capability catered toward solving a particular use case. There is a plethora of Routines housed within SensibleAI Studio and more to come in future iterations and updates to the solution. All available Routines are found in the Explore page within the Routine Library.

> **Example:** Some examples of Routines include Forecast Allocation, Source Data Analysis, ML Regression, K-Means Clustering, and so on.

Routine Instance

A user-created routine object that is required to execute a Routine method. Routine Instances are created from both the Explore and Run pages.

Routine Method

A discrete, executable function that is specific to a given Routine. Routine methods are run by using the Invoke or Rerun buttons on the Run page.

Routine Constructor

A function specific to a given Stateful Routine that encapsulates logic to be executed before any additional Routine Methods can be invoked. Constructor Runs are automatically triggered upon the creation of a Routine Instance and may only be executed once per instance. Although Constructor Runs appear in the Routine Runs section of the Run page alongside other Routine Method Runs, certain actions such as Rerun and Delete are prohibited for Constructor Runs.

Memory Capacity

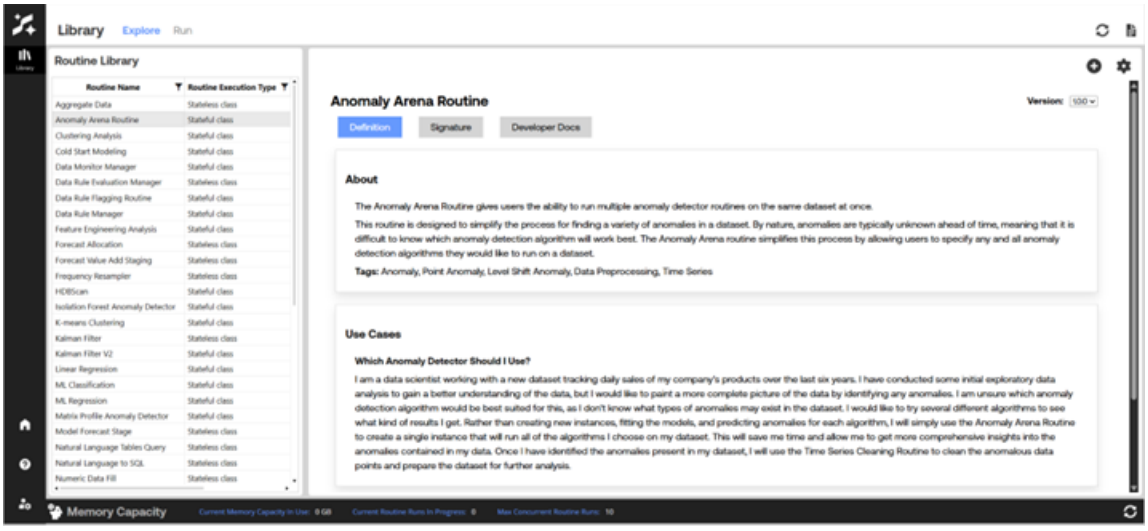
- The amount of memory, in gigabytes, that a Routine Run may use. Users may configure memory capacity as needed, but Routine Methods default to 2GB.
- Memory Capacity Rules
    - SensibleAI Studio users can configure the memory capacity levels. The following list outlines the order of precedence for determining the memory capacity of a Routine Run:
        - The user-defined memory capacity set during the creation of the Routine Run, if it was set.
        - The user-defined memory capacity specifically set on the Routine Method, if it was set.
        - The routine developer-defined memory capacity for that specific Routine Method, if it was set.
        - The user-defined memory capacity set during creation of the Routine Instance, if it was set.

- The user-defined defaults memory capacity for the specific Routine, if it was set.
- The routine developer-defined memory capacity for that specific Routine, if it was set.
- Default to the global default routine memory if none of the above were set.

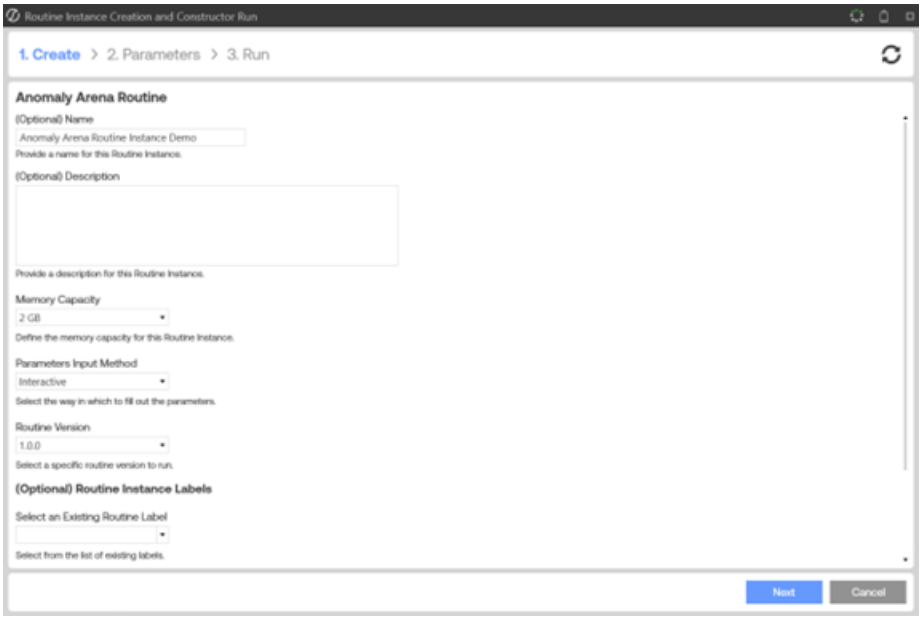
Stateful Routine

A Stateful Routine is a routine that maintains internal state over time. These routines keep track of internal data that may be updated when its Routine Methods are invoked. Stateful Routines will typically require input parameters when creating a Routine Instance.

> **Example:** Anomaly Arena Routine maintains a list of models fit to the provided input data. Below shows how to navigate the component workflow for a Stateful Routine.



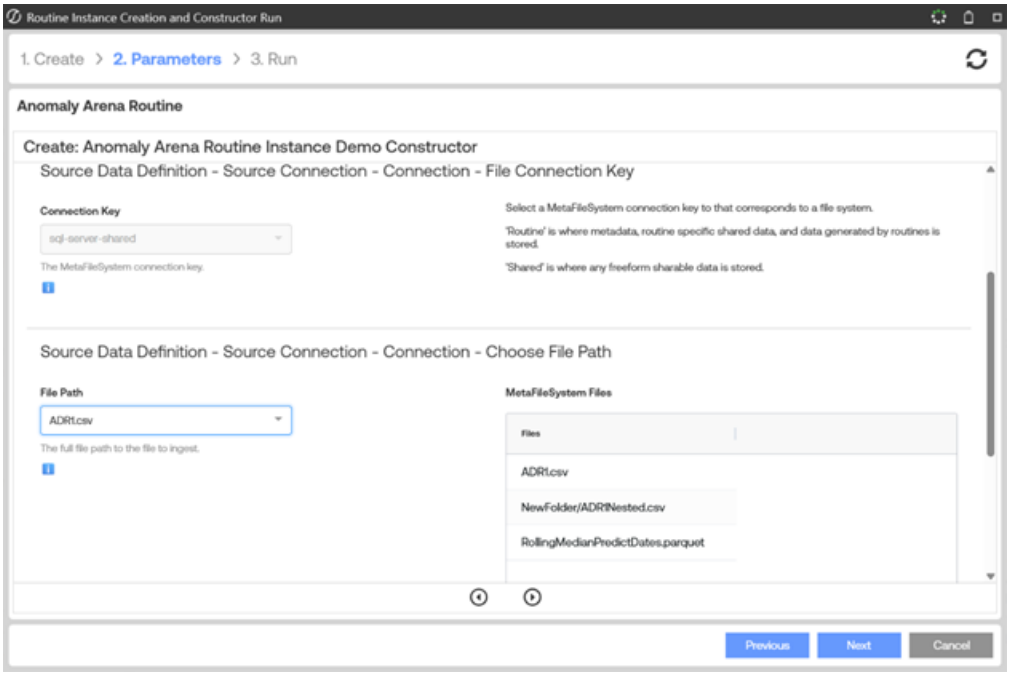
Select your Routine from the list of available Routines. For this example, **The Anomaly Arena Routine** is used. Click **Create** at the top right of the screen.



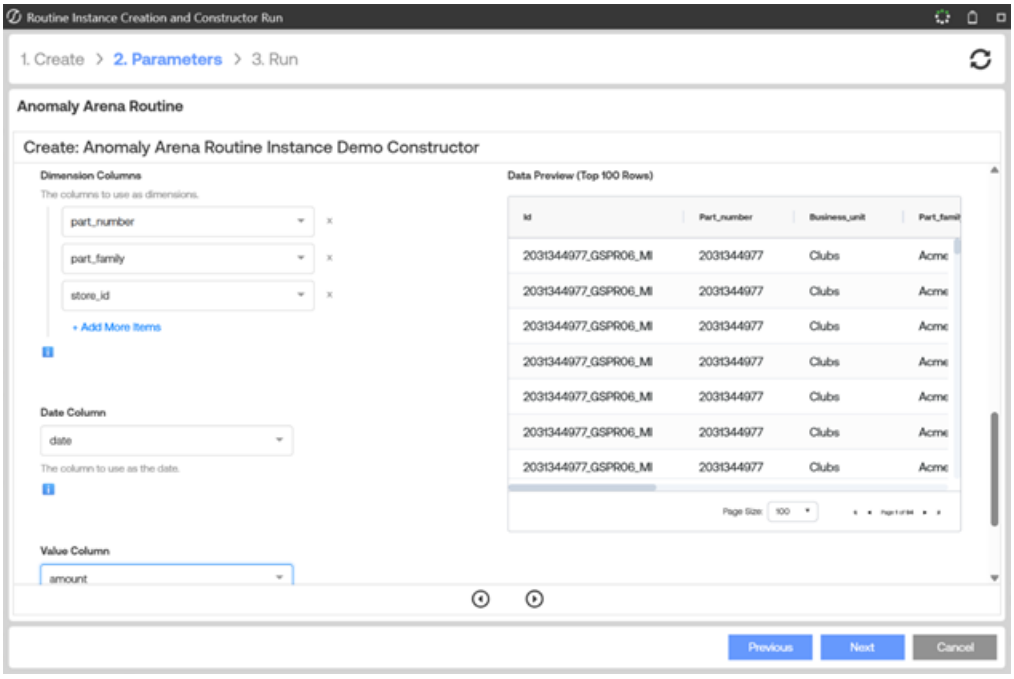
Name your Routine Instance, update memory capacity if necessary, and choose the desired Routine Version. The latest version will always be selected by default.

> **NOTE:** The default size is 2GB. Increase memory capacity size for larger datasets. Scroll down.

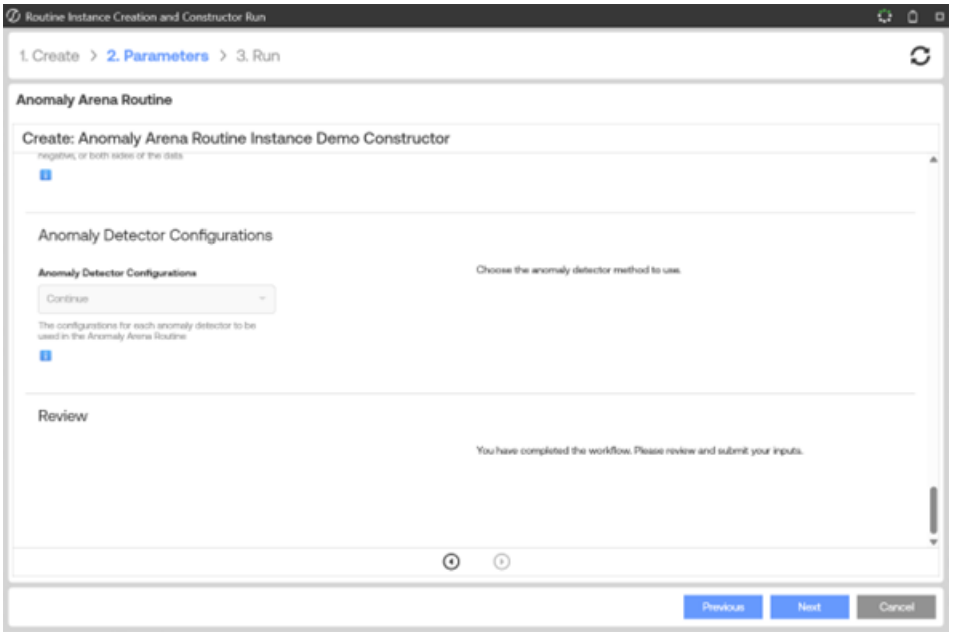
Select one or more existing labels for your Routine Instance. To create a new label, enter the desired name in the text box and click the Create button to the right of it. All newly created or selected labels will show next to the Routine Labels text on the last line. To remove a newly created label, ensure the name is correctly spelled in the text box and click the Delete button to the right of the text box. To remove and existing label that has been selected, simply uncheck it from the dropdown menu. To delete all labels, click the Delete button on the last line. Click **Next**.



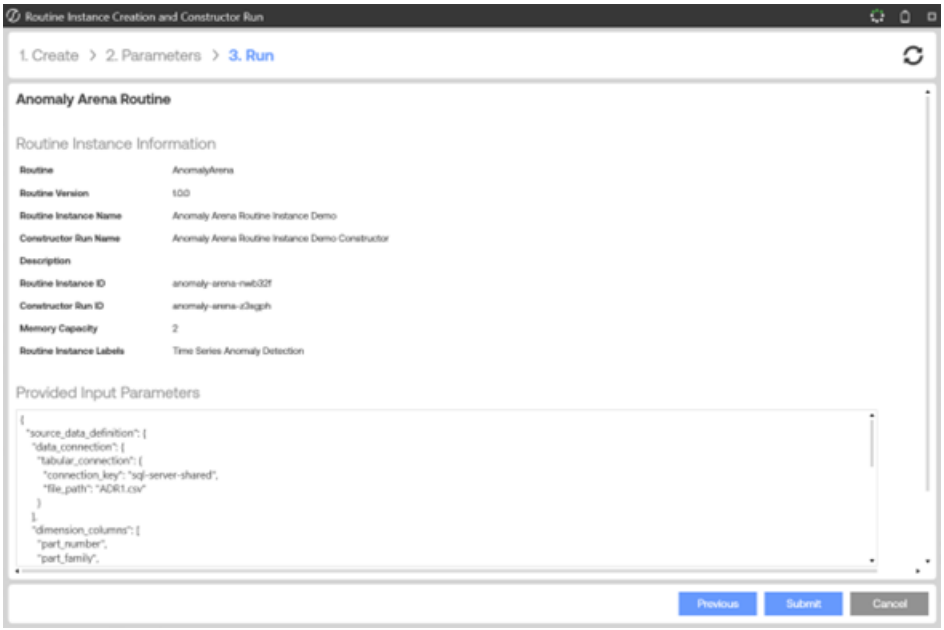
Navigate through the Component Workflow while making selections based on the specific needs of your use case. Many steps may be left to default configurations, but for steps such as **Table Selection**, users need to select their specific table from the drop-down menu.



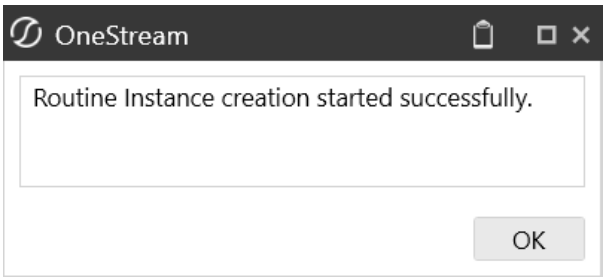
Continue to complete each step of the Component Workflow, selecting **Dimension**, **Date**, and **Value** columns that align with your selected dataset. If adding in a **Feature** dataset, ensure to select **Add** from the drop-down menu. If you are not adding features, select **Continue**. Ensure to include a **Group Name** when prompted.



Update selections to configure one or more anomaly detectors. Once finished with adding anomaly detectors, select **Continue** from the drop-down menu to move on.



After completing the Component Workflow, click **Next** to review the details of your Routine Instance. To make adjustments, click **Previous** to navigate back to the desired step. To create your Routine Instance, click **Submit** on the final Review step. A notification will pop up notifying you that Routine Instance creation has begun. You can view your Routine Instance under the Routine Instances panel on the Run page. The associated Constructor Run will also be visible under Routine Runs.



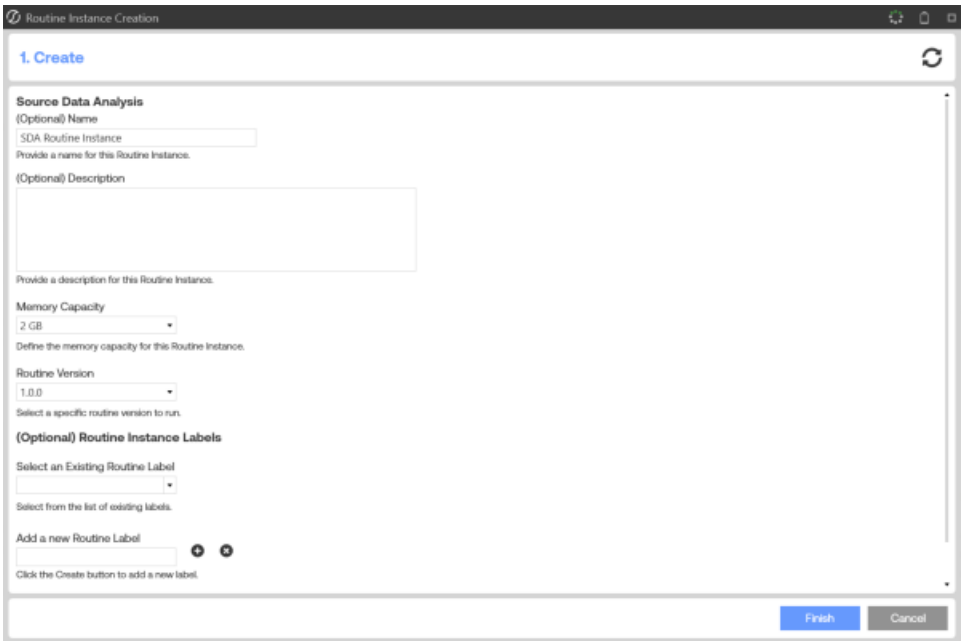
> **NOTE:** The Component Workflow for Stateful Routines will vary based on the specific routine selected; each has its own configurations specific to the type of Routine. Inputs may differ slightly from what has been shown in this walkthrough.

Stateless Routine

A routine that does not maintain any sort of internal state. Stateless Routine's methods can be described as static. Unlike Stateful Routines, Stateless Routines do not require a constructor do not require any input parameters when creating a Routine Instance.

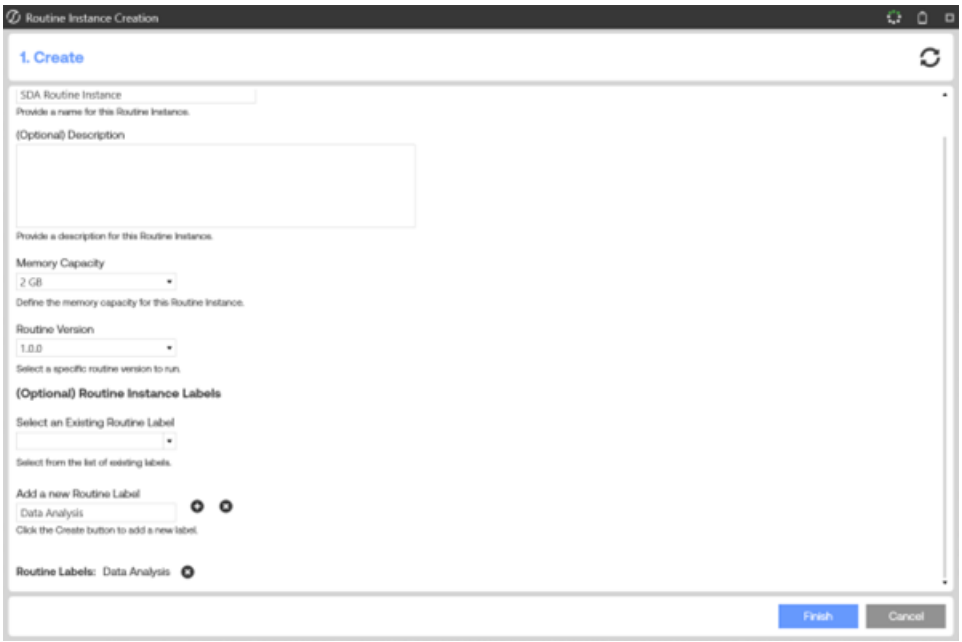
> **Example:** Source Data Analysis – does not require input parameters; no internal state.

Navigate to the **Explore** page of SensibleAI Studio to begin this process. Select a routine. For this example, the **Source Data Analysis Routine** will be used. Click **Create** at the top right of the screen.



Name your Routine Instance and include a description (optional). Ensure that the correct Routine Version is selected.

> **NOTE:** Due to the nature of this being a Stateless Routine, there is no constructor method that will be run upon Routine Instance creation. Therefore, there will **not** be a Component Workflow.



Select one or more existing labels for your Routine Instance. To create a new label, enter the desired name in the text box and click the Create button to the right of it. All newly created or selected labels will show next to the Routine Labels text on the last line. To remove a newly created label, ensure the name is correctly spelled in the text box and click the Delete button to the right of the text box. To remove and existing label that has been selected, simply uncheck it from the dropdown menu. To delete all labels, click the Delete button on the last line.

After clicking the **Submit** button, click on the **Run** page at the top of the screen to locate and interact with this Routine Instance.

Routine Instances

All Routine Instances are listed in the Routine Instances table on the Run page. To view only the instances you have created, click the **user-icon** ( ).

When you select a Routine Instance, the Routine Runs table will automatically update to show only the runs associated with that instance. To clear your selection, click the **deselect icon** ( ), this will also remove the filter from the Routine Runs table and display all runs.

Routine Instance Actions

When a Routine Instance has been selected on the Run page, the following actions may be completed: Create, Invoke, Edit, or Delete.

Create

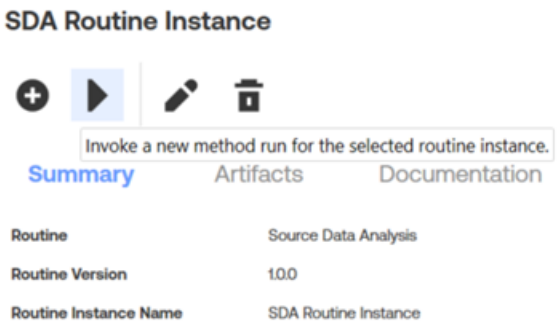
The Create button allows users to generate a new Routine Instance based on the selected Routine Type. It works the same as the **Create** button on the **Explore** page. For detailed instructions on creating Routine Instances, see Stateless Routine and Stateful Routine sections.

Invoke

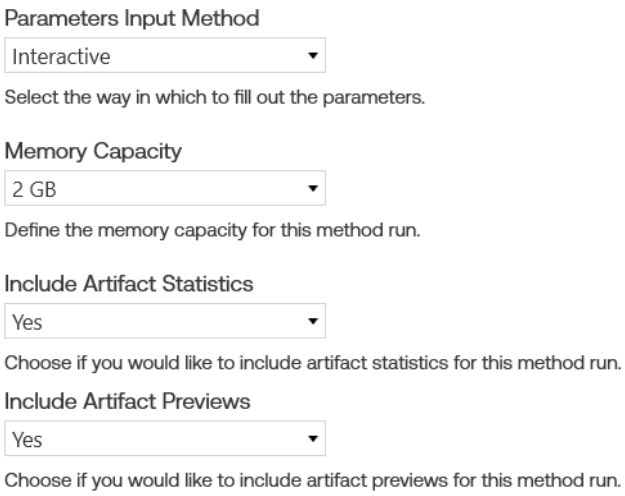
The **Invoke** button starts a new Routine Method Run for the selected Routine Instance.

> **Example:** For this example, a Routine Instance for **Source Data Analysis** is used.

1. Once the Routine Instance is selected, click the Invoke button on the right.



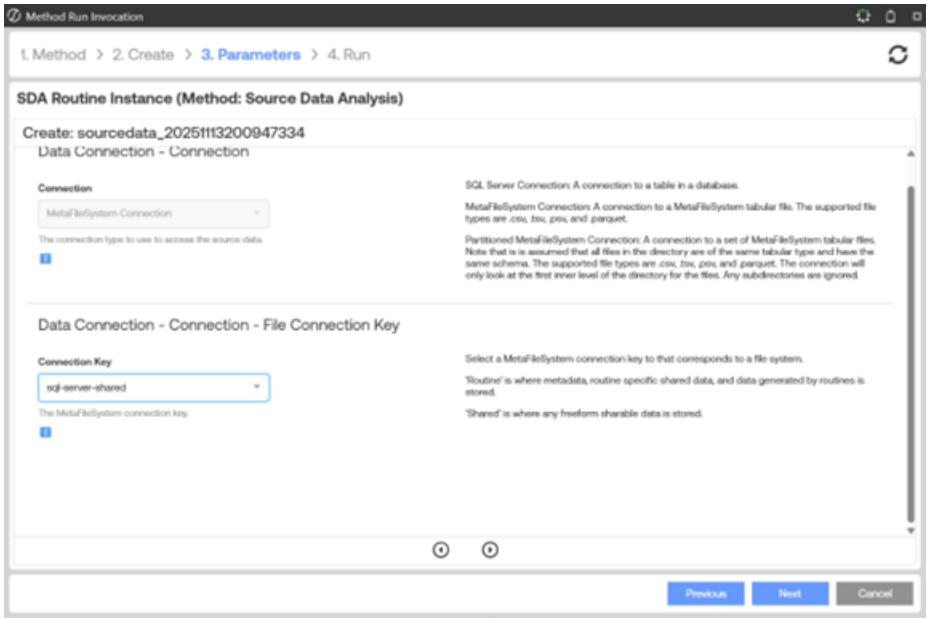
2. Select the **Source Data Analysis** method then click **Next**.
3. Name your Routine Run and ensure that the **Input Method** is set to **Interactive**. Artifact statistics and previews will automatically be generated by default; set either option to No if you do not want statistics or previews to be generated.

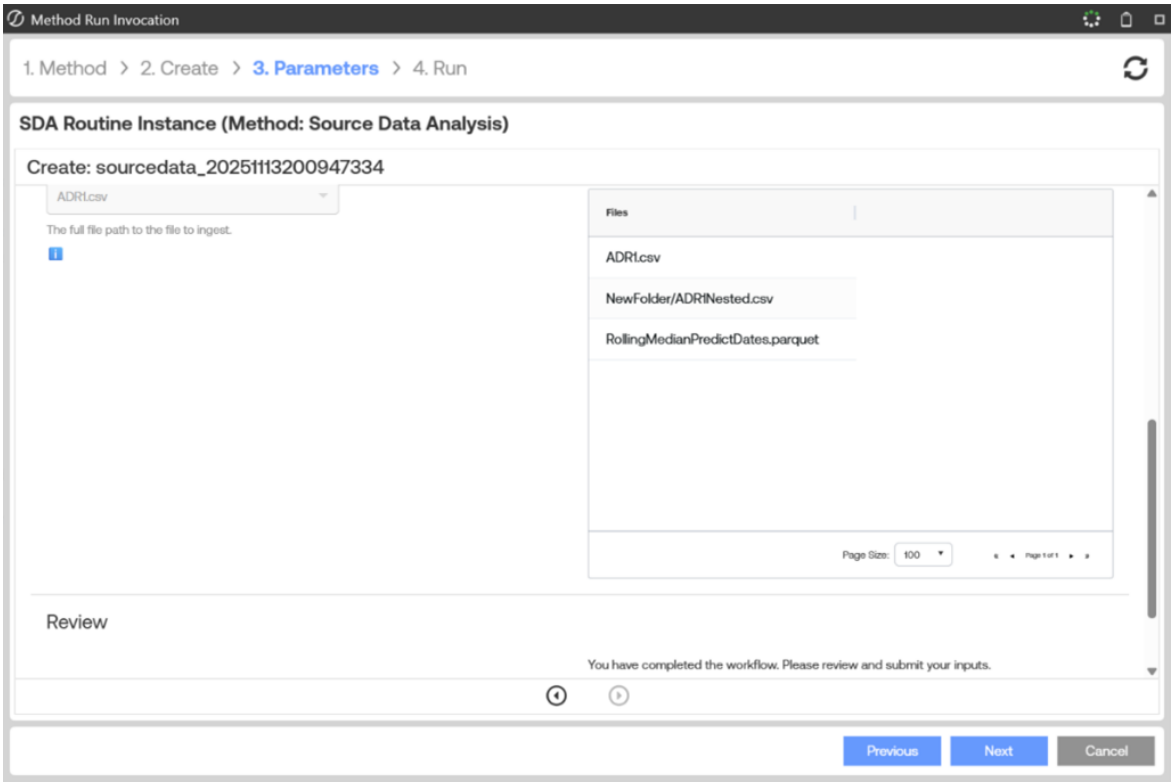


> **NOTE:** Setting the Input Method to Interactive will task users with completing a Component Workflow process.

4. Select or create new labels for the Routine Method Run and click **Next**.

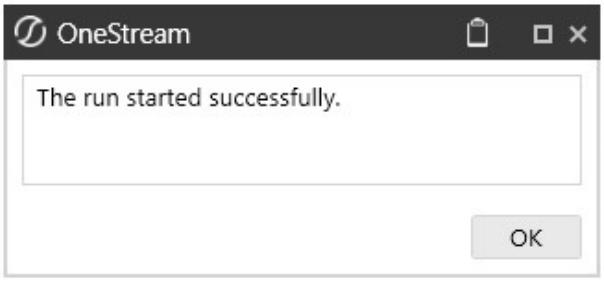
5. Complete the interactive workflow process. These steps follow the same logic as with other Component Workflow tasks; users need to select a **Database Connection**, a **Database Resource** or **File Connection Key**, and a **Database Name** or **File Path** to start.





Once you hit the Review step, you have completed the Component Workflow. Click **Next**.

6. Review input parameters and details of your Source Data Analysis Routine Method Run and click **Submit** to start the run.



After submitting, it may take a few minutes for the run to complete.

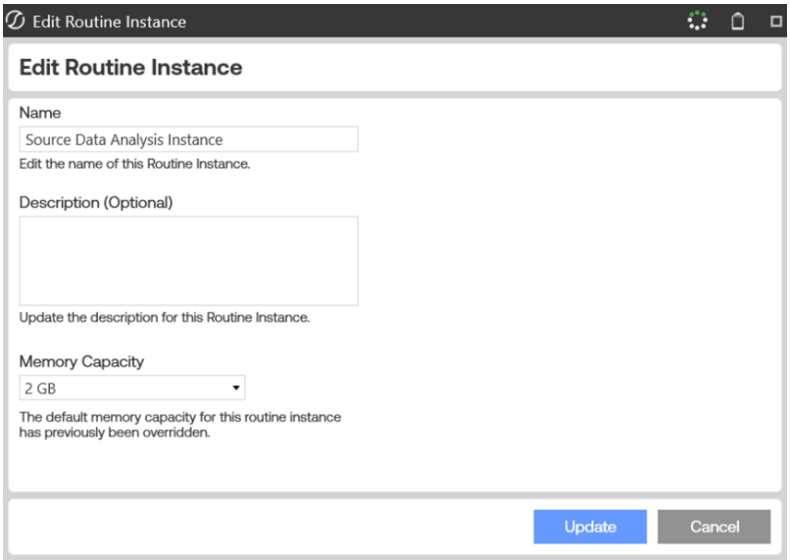
> **NOTE:** The Component Workflow will vary based on the routine method selected. Inputs may differ slightly from what has been shown in this walkthrough.

Additionally, Json inputs may be used to run a new method in place of the Component Workflow by setting the **Input Method** to Json when creating a new method run. After completing the Component Workflow for a given method, the Json input can be copied on the Review page before executing the run. After the run has started, this Json input can also be accessed by clicking the Copy button.

Edit

Users update the Routine Instance Name and Routine Instance Description by clicking the **Edit** button.

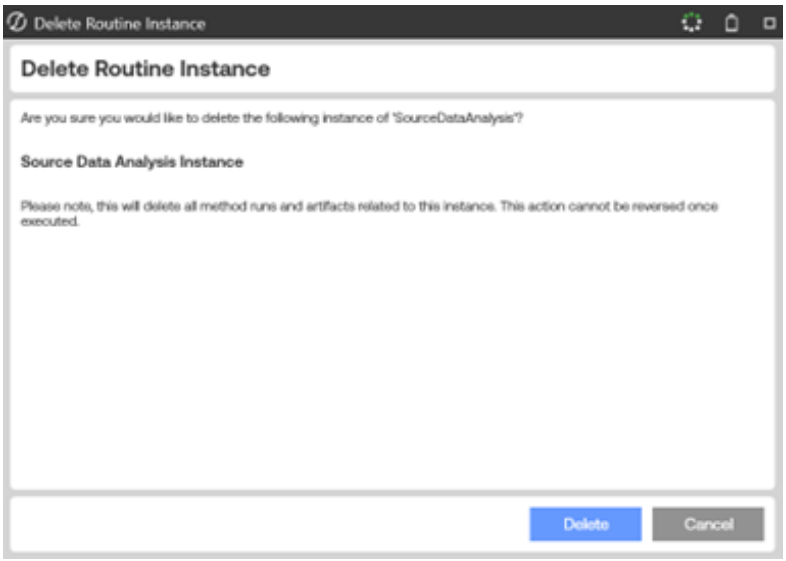
> **NOTE:** It is intended that the Routine Run Name and Routine Run Description do not change.



Delete

Users delete selected routine instances entirely by clicking the **Delete** button.

> **CAUTION:** All Routine Method Runs and Artifacts attached to this Routine Instance will also be deleted. All deletions cannot be reversed.



Routine Runs

A Routine Run is the term used to describe any routine-specific job that is executed by a user.

Each row in the Routine Runs table, located on the Run page of SensibleAI Studio, represents a Routine Run. This includes **Constructor Runs**, which are triggered immediately after the successful creation of a Routine Instance, as well as **Method Runs**, which are specific to a particular Routine Instance and vary depending on the Routine. Available actions depend on whether a Routine Constructor Run, or a Routine Method Run has been selected.

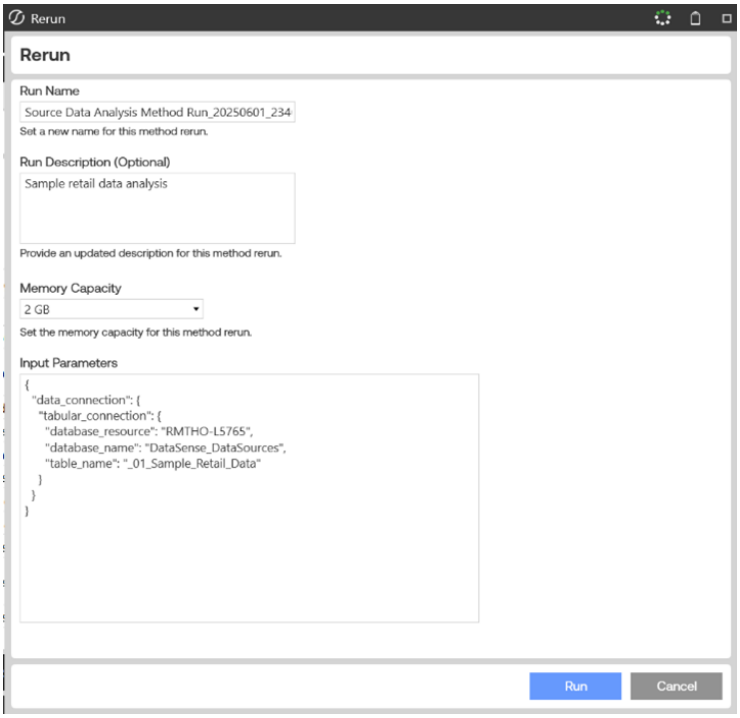
To view only the runs you have created, click the **user-icon** ( ). Additional filters for Routine Runs such as Stateful, Stateless, Today, Completed, Running, and Queued can be applied individually or in combination to further customize the view. To clear all Routine Run filters, click the **clear filters icon** ( ).

Routine Run Actions

When a Routine Method Run has been selected, the following actions may be completed: Rerun, Copy, Edit, Delete, or Cancel.

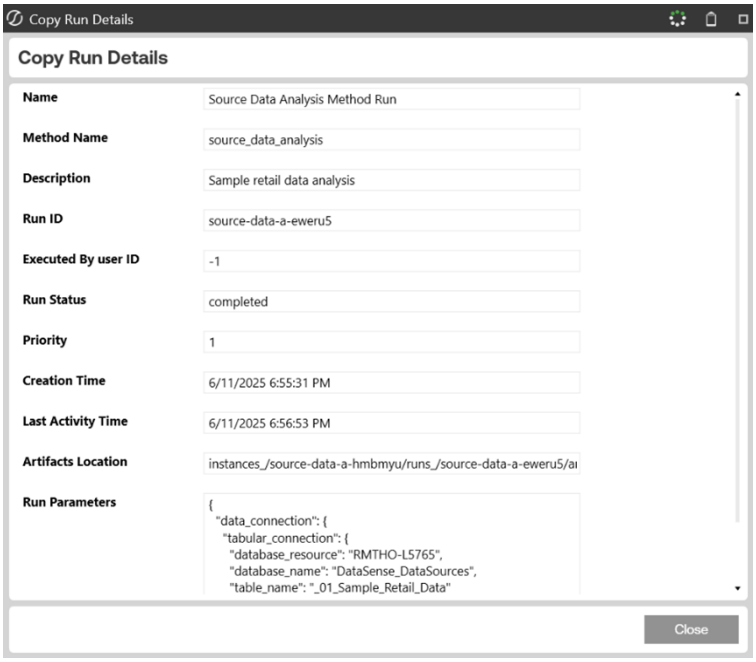
Rerun

Users choose to rerun their selected Routine Method Run by clicking the **Rerun** button. The system will automatically update the Routine Run Name by appending the current date and time to the original name. Users can modify input parameters by editing the provided JSON snippet, and optionally adjust the memory capacity or description for this new run.



Copy

Users copy a selected run's details by clicking the **Copy** button. Copy the Json Input Parameters to execute a new Routine Method Run by setting the **Input Method** to Json instead of Interactive when configuring your new method run.



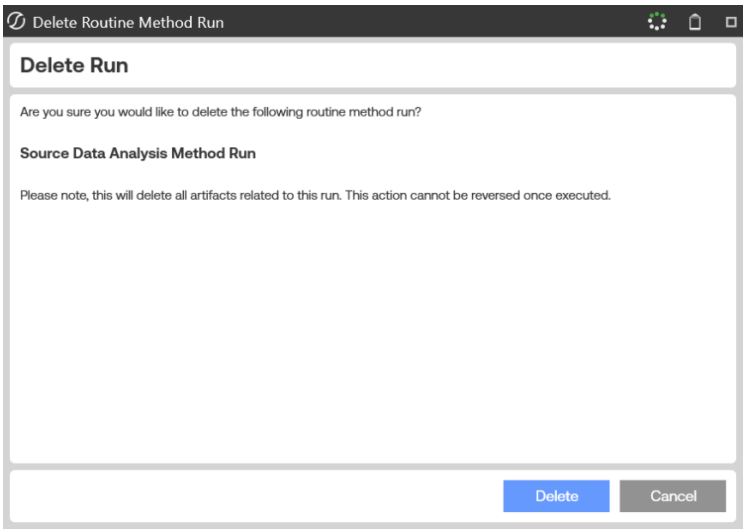
Edit

Users update the name and description of a Routine Method Run by clicking the **Edit** button.

Delete

Users delete a selected Routine Method Run by clicking the **Delete** button.

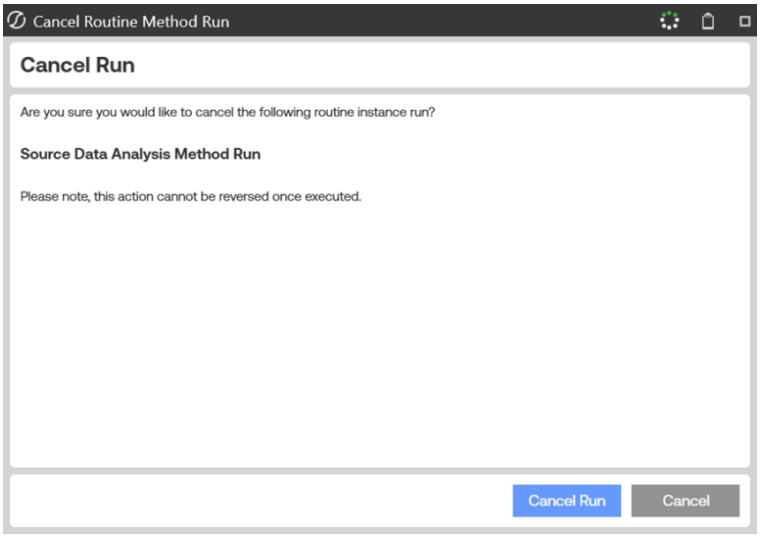
> **CAUTION:** All artifacts generated from this Routine Method Run will also be deleted. All deletions cannot be reversed.



Cancel

Users cancel a run once it has started by clicking the **Cancel** button.

> **NOTE:** This action cannot be executed if a run has already completed successfully or entered an error state.

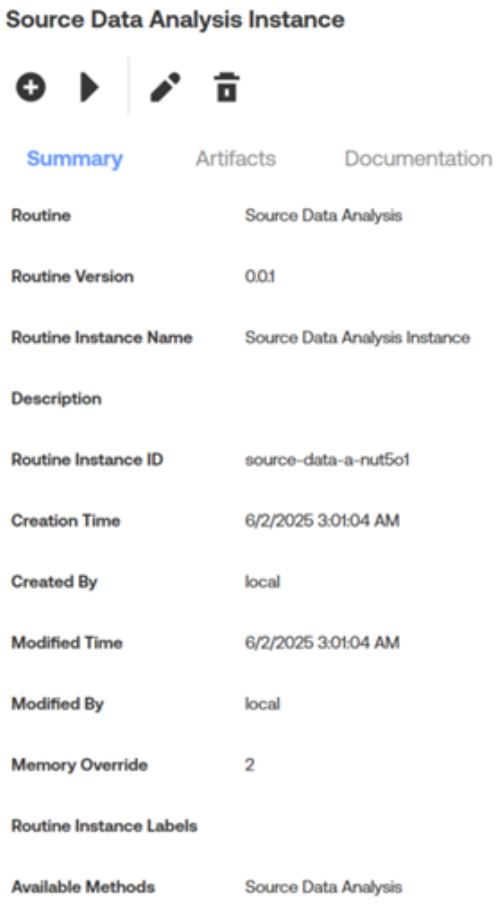

Cancel Routine Method Run
Cancel Run
Are you sure you would like to cancel the following routine instance run?
Source Data Analysis Method Run
Please note, this action cannot be reversed once executed.
Cancel Run
Cancel

Routine Instance and Routine Run Views

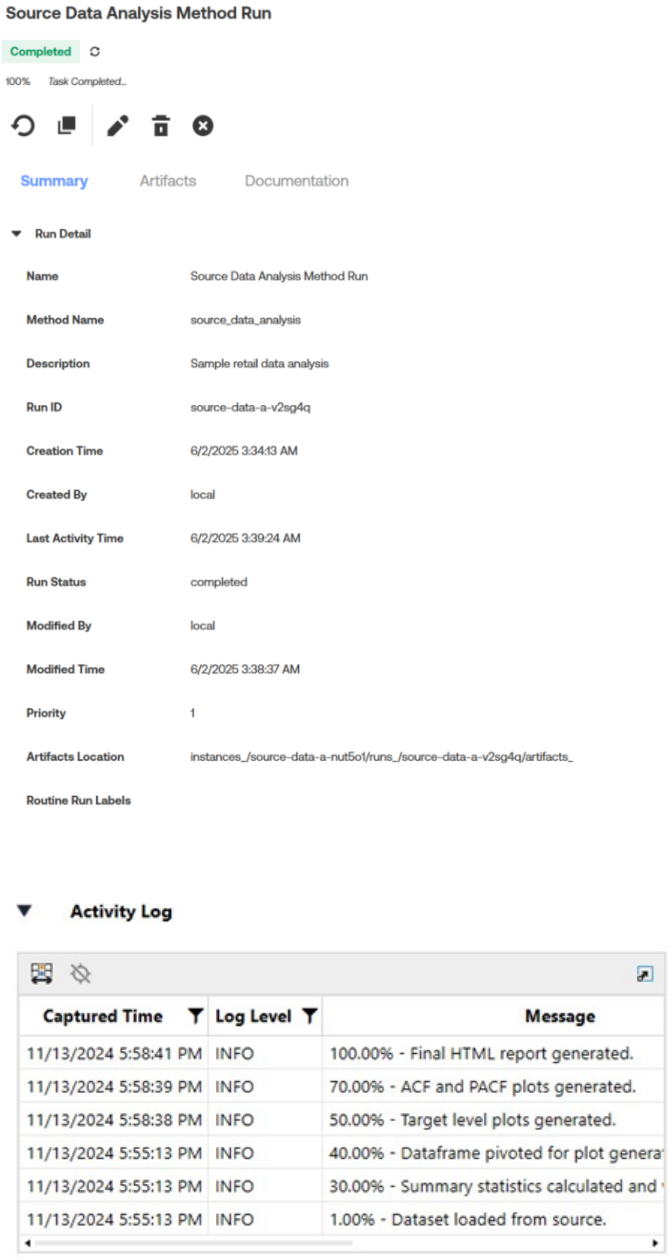
The following views are available for both Routine Instances and Routine Runs: Summary, Artifacts, and Documentation.

Summary

If a Routine Instance has been selected, the Summary view shows additional details about the instance including ID, Creation Time, Created By, and Available Methods.



If a Routine Run has been selected, users view a summary of the run, including details and a log of run-specific activities. A drop-down menu is provided to access Run Details and the Activity Log.


Source Data Analysis Method Run
Completed
100% Task Completed...
Summary
Artifacts
Documentation
Run Detail
Name
Source Data Analysis Method Run
Method Name
source_data_analysis
Description
Sample retail data analysis
Run ID
source-data-a-v2sg4q
Creation Time
6/2/2025 3:34:13 AM
Created By
local
Last Activity Time
6/2/2025 3:39:24 AM
Run Status
completed
Modified By
local
Modified Time
6/2/2025 3:38:37 AM
Priority
1
Artifacts Location
instances_/source-data-a-nut5o1/runs_/source-data-a-v2sg4q/artifacts_
Routine Run Labels

Activity Log
Captured Time
Log Level
Message
11/13/2024 5:58:41 PM INFO 100.00% - Final HTML report generated.
11/13/2024 5:58:39 PM INFO 70.00% - ACF and PACF plots generated.
11/13/2024 5:58:38 PM INFO 50.00% - Target level plots generated.
11/13/2024 5:55:13 PM INFO 40.00% - Dataframe pivoted for plot genera
11/13/2024 5:55:13 PM INFO 30.00% - Summary statistics calculated and
11/13/2024 5:55:13 PM INFO 1.00% - Dataset loaded from source.

Artifacts

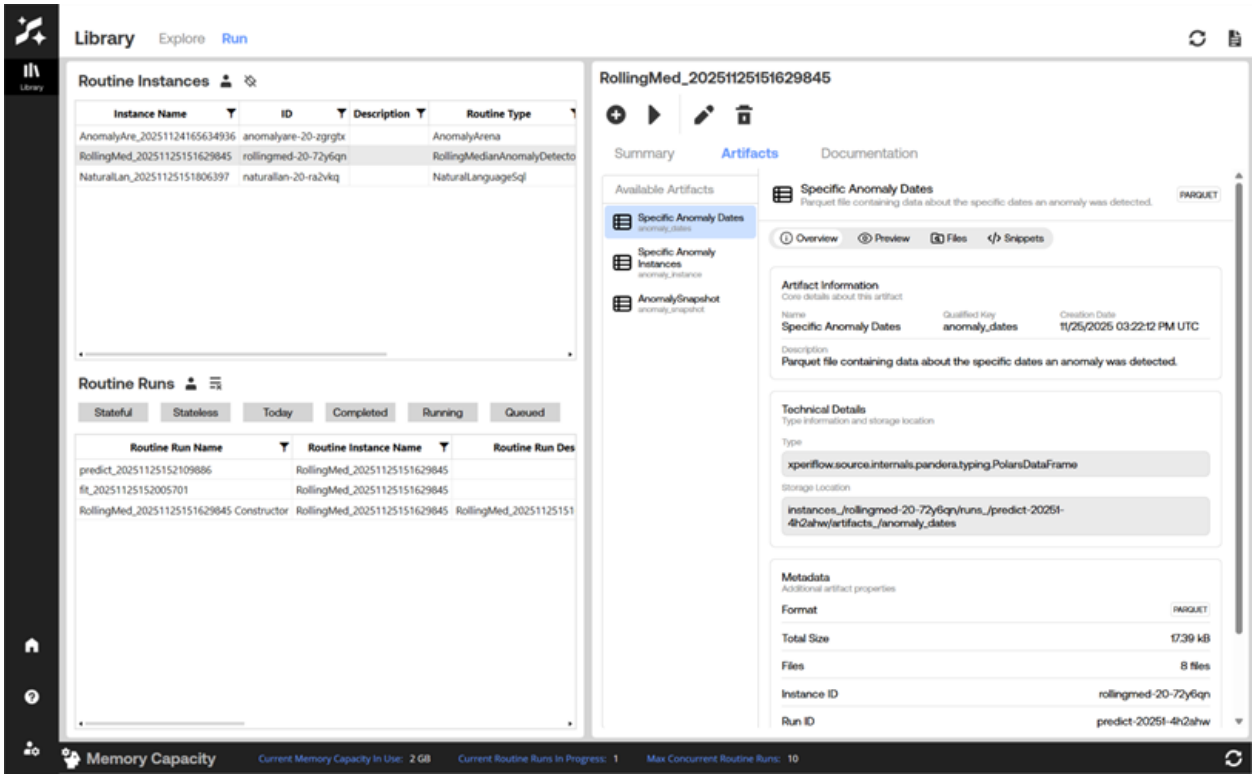
Once a Routine Method Run has successfully completed, view the generated data outputs by clicking on the **Artifacts** tab. An Artifact is a file-based data output that is generated by successfully executing a Routine Method Run. All Methods of a Stateless Routine will produce at least one Artifact; certain Methods of Stateful Routines may produce zero Artifacts. Examples of Artifact output types include parquet files, HTML files, and PDFs.

> **NOTE:** If a user has selected a Routine Instance rather than a Routine Method Run, the Artifacts view will display artifacts from all routine method runs associated with that routine instance.

Using our Time Series Data Analysis example, to view an Artifact:

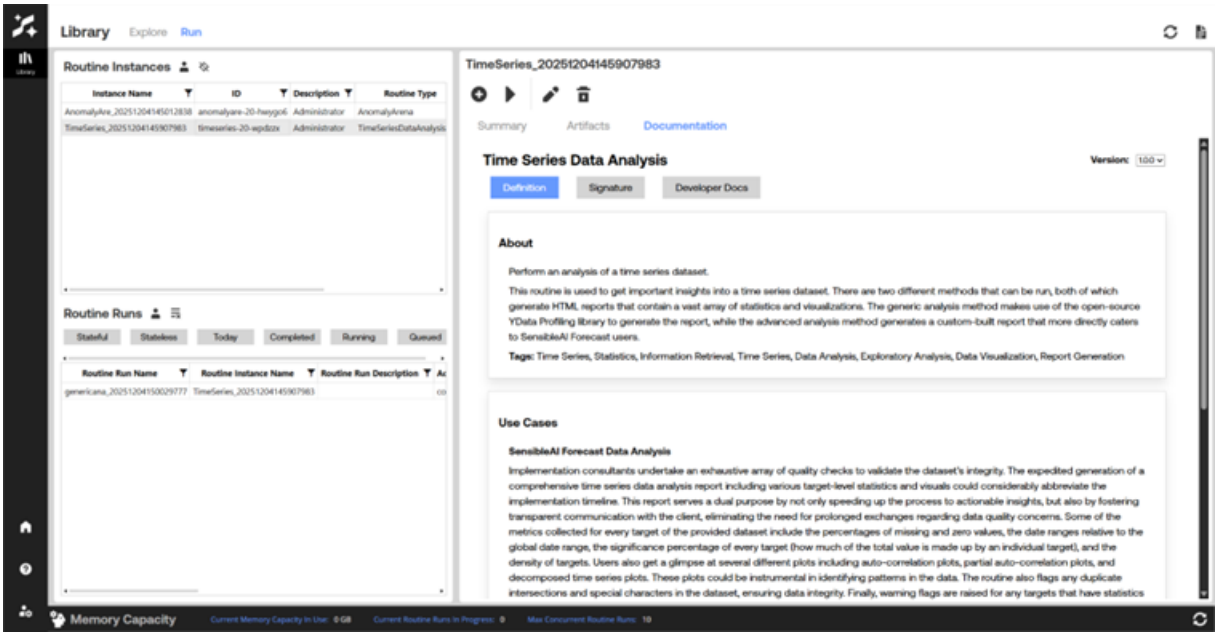
1. Click on the Routine Method Run from the Routine Run Activity table.
    a. Wait for it to complete. Monitor the activity status and refresh until the method is completed – may take several minutes.
    b. If a syserror occurs, refer to the **Activity Log** drop-down menu within the Summary section on the right-hand side of the page. It is possible there is a misconfiguration made in certain column selections when stepping through the component workflow. Repeat steps above as needed.
2. When the routine run completes, click on the **Artifacts** button on the right-hand side of the page.
    a. Navigate to the artifact named Generic or Advanced Time Series Report.
    b. Within the files section, click the file called html_content.html.

A box appears containing the actual HTML file.



Documentation

The documentation section contains the same Routine Documentation as the Explore page, which is dependent on the Routine Instance or the Routine Run that is selected.


Library
Explore
Run
Routine Instances
Instance Name
ID
Description
Routine Type
Routine Runs
Stateful
Stateless
Today
Completed
Running
Queued
Routine Run Name
Routine Instance Name
Routine Run Description
Memory Capacity
TimeSeries_20251204145907983
Summary
Artifacts
Documentation
Time Series Data Analysis
Version:
Definition
Signature
Developer Docs
About
Perform an analysis of a time series dataset.
This routine is used to get important insights into a time series dataset. There are two different methods that can be run, both of which generate HTML reports that contain a vast array of statistics and visualizations. The generic analysis method makes use of the open-source YData Profiling library to generate the report, while the advanced analysis method generates a custom-built report that more directly caters to SensibleAI Forecast users.
Tags: Time Series, Statistics, Information Retrieval, Time Series, Data Analysis, Exploratory Analysis, Data Visualization, Report Generation
Use Cases
SensibleAI Forecast Data Analysis
Implementation consultants undertake an exhaustive array of quality checks to validate the dataset's integrity. The expedited generation of a comprehensive time series data analysis report including various target-level statistics and visuals could considerably abbreviate the implementation timeline. This report serves a dual purpose by not only speeding up the process to actionable insights, but also by fostering transparent communication with the client, eliminating the need for prolonged exchanges regarding data quality concerns. Some of the metrics collected for every target of the provided dataset include the percentages of missing and zero values, the date ranges relative to the global date range, the significance percentage of every target (how much of the total value is made up by an individual target), and the density of targets. Users also get a glimpse at several different plots including auto-correlation plots, partial auto-correlation plots, and decomposed time series plots. These plots could be instrumental in identifying patterns in the data. The routine also flags any duplicate intersections and special characters in the dataset, ensuring data integrity. Finally, warning flags are raised for any targets that have statistics

Package Specifications and Usage Limits

There are two packages available for SensibleAI Studio: Essentials and Standard.

SensibleAI Studio "Essentials" Package

Description

The “Essentials" Package provides AI tools and capabilities to accelerate and enhance your experience around the usage of SensibleAI Forecast and AI Powered Planning processes.

Offering Includes

- SensibleAI Studio Solution
- SensibleAI Studio “Essentials” Routines SensibleAI Studio “Essentials” Solutions
    - Ex: AI Data Tools
    - Ex: AI Time Series Anomaly Detector
- “Essentials” Genesis Blocks
    - Ex: Genesis Block - AI Routine Runner

Usage Limits

Definition	Interaction Locations
A maximum of 3000 Invocations per calendar quarter of LLM Chat Models surface through SensibleAI Studio and Xperiflow Business Rules.	• Xperiflow Business Rules: XperiflowAIClient.PostLLMChatResourceAsync • Xperiflow Business Rules: Xperiflow.LanguageModels.LLMChatClient.InteractAsync
A maximum of 300,000 LLM Embedding Generations per quarter.	• Xperiflow Business Rules: XperiflowAIClient.PostLLMEmbeddingResourceAsync • Xperiflow Business Rules: Xperiflow.LanguageModels.LLMEmbeddingClient.GenerateEmbeddingsAsync • Xperiflow Business Rules: Xperiflow.LanguageModels.LLMEmbeddingClient.GenerateEmbeddingAsync • SensibleAI Studio: “Text Embeddings” Routine

SensibleAI Studio "Standard" Package

Description

The “Standard” Package includes all SensibleAI Studio “Essentials” capabilities as well as additional SensibleAI Studio Powered Solutions, and Genesis Blocks.

Offering Includes

- SensibleAI Studio Solution
- SensibleAI Studio “Standard” and “Essentials” Routines SensibleAI Studio Powered “Standard” and “Essentials” Solutions
- Ex: AI Data Tools
    - Ex: AI Time Series Anomaly Detector
    - Ex: AI System Diagnostics
    - Ex: AI Account Recs
- “Standard” and “Essentials” AI Genesis Blocks
    - Ex: AI Genesis Block – AI Commentary Summarizer
    - Ex: AI Genesis Block – AI Routine Runner
    - Ex: AI Genesis Block – AI NL to Table

Usage Limits

Definition	Interaction Locations
A maximum of 3000 Invocations per calendar quarter of LLM Chat Models surface through SensibleAI Studio and Xperiflow Business Rules.	• Xperiflow Business Rules: XperiflowAIClient.PostLLMChatResourceAsync • Xperiflow Business Rules: Xperiflow.LanguageModels.LLMChatClient.InteractAsync

Definition	Interaction Locations
A maximum of 300,000 LLM Embedding Generations per quarter.	• Xperiflow Business Rules: XperiflowAIClient.PostLLMEmbeddingResourceAsync • Xperiflow Business Rules: Xperiflow.LanguageModels.LLMEmbeddingClient.GenerateEmbeddingsAsync • Xperiflow Business Rules: Xperiflow.LanguageModels.LLMEmbeddingClient.GenerateEmbeddingAsync • SensibleAI Studio: “Text Embeddings” Routine
A maximum of 100,000 Code Scan Units per quarter.	• SensibleAI Studio: “Code Scanner” Routine • AI System Diagnostics

Definitions

Routine: A modular software component comprised of one or more functions that accept inputs and produce outputs, such as data results or interactive web applications.

SensibleAI Studio Powered Solution: Any OneStream Solution that has integrated one or more Routines from SensibleAI Studio.

Invocation: A single call or execution of a service, function, or model within the platform.

LLM Embedding Generation: A single operation where an LLM embedding model processes one discrete sub-set of text to produce its numeric embedding representation.

Scan Item: A specific code file or script selected for analysis within a code scanning process.

Condition: A predefined coding pattern or issue (e.g., "Unnecessary Logging Statements", "Nested SQL loops") that the scanning process searches for within a Scan Item.

Code Scan Unit: A single evaluation that checks a Scan Item against one defined Condition, returning any matching instances within that file.

Help and Miscellaneous Information

Display Settings

OneStream Solutions frequently require the display of multiple data elements for proper data entry and analysis. Therefore, the recommended screen resolution is a minimum of 1920 x 1080 for optimal rendering of forms and reports.

Additionally, OneStream recommends that you adjust the Windows System Display text setting to 100% and do not apply any Custom Scaling options.

Package Contents and Naming Conventions

The package file name contains multiple identifiers that correspond with the platform. Renaming any of the elements contained in a package is discouraged in order to preserve the integrity of the naming conventions.

Example Package Name: STU_SV120_XPFv4.1.0_PV910_PackageContents.zip

Identifier	Description
STU	Solution ID
SV120	Solution Version

Identifier	Description
XPFv4.1.0	Xperiflow Version
PV910	Minimum Platform version required to run solution
PackageContents	File name

OneStream Solution Modification Considerations

A few cautions and considerations regarding the modification of OneStream Solutions:

- Major changes to business rules or custom tables within a OneStream Solution will not be supported through normal channels as the resulting solution is significantly different from the core solution.
- If changes are made to any dashboard object or business rule, consider renaming it or copying it to a new object first. This is important because if there is an upgrade to the OneStream Solution in the future and the customer applies the upgrade, this will overlay and wipe out the changes. This also applies when updating any of the standard reports and dashboards.
- If modifications are made to a OneStream Solution, upgrading to later versions will be more complex depending on the degree of customization. Simple changes such as changing a logo or colors on a dashboard do not impact upgrades significantly. Making changes to the custom database tables and business rules, which should be avoided, will make an upgrade even more complicated.