



Platform **User Guide**

Version: 9.7

Doc Build Date: 12/31/2022

Disclaimers

Except as otherwise provided in an express written agreement, Alteryx Inc. (“ Alteryx”) makes no representations or warranties with respect to the software and documentation contained herein and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Furthermore, Alteryx reserves the right to revise the software and documentation from time to time without the obligation of Alteryx to notify any person of such revisions or changes.

Copyright and Trademark Notices

© 2013 - 2022 Alteryx Inc. All Rights Reserved.

Alteryx Inc.

17200 Laguna Canyon Road

Irvine, CA 92618

Phone: +1 888 836 4274

Alteryx and Trifacta are registered trademarks of Alteryx Inc.

Credits

For third-party license information, please select **About Trifacta** from the Resources menu.

See also [Third-Party License Information](#).

1. Platform	6
1.1 Plans	7
1.1.1 Plan Basics	8
1.1.2 Overview of Orchestration	11
1.1.3 Plan View Page	15
1.1.3.1 Plan Run Details Page	18
1.1.3.2 Share Plan Dialog	20
1.1.3.3 Manage Plan Notifications Dialog	21
1.1.3.4 Manage Parameters Dialog for Plans	23
1.1.3.5 Plan View for Flow Tasks	24
1.1.3.6 Plan View for HTTP Tasks	30
1.1.3.7 Plan View for Slack Tasks	32
1.1.3.8 Plan View for Delete Tasks	34
1.1.3.9 Plan View for Triggers	36
1.1.4 Plans Page	39
1.1.5 Plan Tasks	41
1.1.5.1 Create a Plan	42
1.1.5.1.1 Create Delete Task	50
1.1.5.1.2 Create HTTP Task	53
1.1.5.1.3 Create Slack Task	63
1.1.5.2 Diagnose Failed Plan Runs	66
1.1.5.3 Export Plan	68
1.1.5.4 Import Plan	69
1.1.5.5 Share a Plan	70
1.1.6 Plan Metadata References	72
1.2 Library	79
1.2.1 Library Page	80
1.2.1.1 Imported Datasets Page	83
1.2.1.2 Dataset Details Page	85
1.2.1.3 Import Data Page	89
1.2.1.3.1 Database Browser	95
1.2.1.3.2 File System Browser	98
1.2.1.3.3 HDFS Browser	100
1.2.1.3.4 S3 Browser	102
1.2.1.3.5 File Import Settings	104
1.2.1.3.6 Table Import Settings	106
1.2.1.4 References Page	107
1.2.1.5 Macros Page	109
1.2.1.5.1 Macro Details Page	111
1.2.2 Library Tasks	113
1.2.2.1 Import a File	114
1.2.2.1.1 Change File Encoding	115
1.2.2.1.2 Remove Initial Structure	116
1.2.2.2 Import a Table	117
1.2.2.2.1 Disable Type Inference	118
1.2.2.3 Create Dataset with Parameters	119
1.2.2.3.1 Parameterize Files for Import	122
1.2.2.3.2 Parameterize Tables for Import	129
1.2.2.4 Export Macro	135
1.2.2.5 Import Macro	136
1.2.2.6 Create Dataset with SQL	138
1.3 Connections	145
1.3.1 Connection Basics	146
1.3.2 Connections Page	148
1.3.2.1 Create Connection Window	152
1.3.2.2 Share Connection Dialog	157
1.3.3 Using Connections	161
1.3.3.1 Using Databases	162
1.3.3.2 Using HDFS	164
1.3.3.3 Using S3	168

1.3.3.4	Using SQL DW	172
1.3.4	Connection Tasks	174
1.3.4.1	Configure Connectivity	175
1.3.4.1.1	Relational Access	178
1.3.4.1.2	Enable Custom SQL Query	181
1.3.4.1.3	Configure JDBC Ingestion	183
1.3.4.1.4	Configure Data Source Caching	187
1.3.4.1.5	Configure Security for Relational Connections	190
1.3.4.1.6	Enable SSO for Relational Connections	194
1.3.4.1.7	Enable OAuth 2.0 Authentication	198
1.3.4.1.8	Configure Connectivity for Amazon RDS	219
1.3.4.1.9	Configure Type Inference	220
1.3.4.1.10	Troubleshooting Relational Connections	223
1.3.4.1.11	Supported Connection Credential Types	225
1.3.4.2	Connect to Data	233
1.3.4.2.1	Share a Connection	235
1.3.5	Connection Types	236
1.3.5.1	Oracle Database Connections	240
1.3.5.2	PostgreSQL Connections	244
1.3.5.3	MySQL Connections	248
1.3.5.4	Microsoft SQL Server Connections	252
1.3.5.5	Google Sheets Connections	256
1.3.5.6	External S3 Connections	258
1.3.5.7	Amazon Redshift Connections	264
1.3.5.8	AWS Glue Connections	271
1.3.5.9	Snowflake Connections	276
1.3.5.10	Azure SQL Database Connections	282
1.3.5.11	Microsoft SQL Data Warehouse Connections	284
1.3.5.12	AlloyDB Connections	288
1.3.5.13	Databricks Tables Connections	292
1.3.5.14	SFTP Connections	299
1.3.5.15	REST API Connections	304
1.3.5.16	Enable Teradata Access	312
1.3.5.17	Teradata Connections	314
1.3.5.18	MongoDB Connections	318
1.3.5.19	Tableau Server Connections	323
1.3.5.20	Salesforce Connections	326
1.3.5.21	SharePoint Connections	331
1.3.5.22	Google Analytics Connections	336
1.3.5.23	DB2 Connections	339
1.4	Job History	342
1.4.1	Job History Page	343
1.4.1.1	Job Details Page	348
1.4.1.1.1	Publishing Dialog	360
1.4.1.1.2	Schema Changes Dialog	362
1.4.1.2	Plan Runs Page	364
1.4.1.3	Sample Jobs Page	368
1.4.2	Diagnose Failed Jobs	371
1.5	Schedules	377
1.5.1	Overview of Scheduling	378
1.5.2	Schedules Page	380
1.5.3	Add Schedule Dialog	382
1.5.4	Schedule a Job	385
1.5.5	Schedule a Plan	387
1.6	Profile	388
1.6.1	Preferences Page	389
1.6.1.1	User Profile Page	391
1.6.1.1.1	Storage Config Page	393
1.6.1.2	Account Settings Page	395

1.6.1.3	Email Notifications Page	397
1.6.1.4	Sessions Page	399
1.6.1.5	Access Tokens Page	400
1.6.1.6	Databricks Settings Page	403
1.6.2	Account Management Tasks	405
1.6.2.1	Change Password	406
1.6.2.2	Configure Your Access to S3	407
1.6.2.3	Transfer Asset Ownership	410
1.7	Admin	412
1.7.1	Admin Tasks	415
1.7.1.1	Operations Tasks	416
1.7.1.1.1	Configure Running Environments	417
1.7.1.1.2	Verify Operations	419
1.7.1.2	Access Management Tasks	421
1.7.1.2.1	Enable Access to S3 and AWS Resources	422
1.7.1.2.2	Insert Trust Relationship in AWS IAM Role	426
1.7.1.3	User Admin Tasks	428
1.7.1.3.1	Create User Account	429
1.7.1.3.2	Configure Password Criteria	431
1.7.1.3.3	Create Admin Account	433
1.7.1.3.4	Manage Users under SSO	435
1.7.1.3.5	Invite Users	437
1.7.1.3.6	Create Role	440
1.7.1.4	Application Management Tasks	445
1.7.1.4.1	Manage Downloading	446
1.7.1.4.2	Manage Schedules	449
1.7.1.4.3	Manage Environment Parameters	451
1.7.2	System Services and Logs	454
1.7.3	Storage Maintenance	463
1.7.4	Backup and Recovery	466
1.7.4.1	Platform Rollback	472
1.7.5	Admin Reference	479
1.7.5.1	Deployment Manager Page	480
1.7.5.2	Admin Console	483
1.7.5.2.1	Users Page	485
1.7.5.2.2	Roles Page	491
1.7.5.2.3	Workspace Settings Page	496
1.7.5.2.4	Admin Settings Page	511
1.7.5.2.5	AWS Settings Page	515
1.7.5.2.6	Environment Parameters Page	518
1.7.5.2.7	OAuth 2.0 Clients Page	520
1.7.5.3	Workspace Admin Permissions	521
1.7.5.4	Admin Download Logs Dialog	523
1.7.5.5	Required AWS Account Permissions	526
1.7.5.6	Privileges and Roles Reference	531
1.8	Platform Reference	534
1.8.1	Supported SQL Syntax	535
1.8.2	Locale Settings	537
1.8.3	Supported Time Zone Values	539
1.8.4	cron Schedule Syntax Reference	555

Platform

This section covers the Designer Cloud powered by Trifacta® platform and the applications and objects hosted in it.

Login

For more information, see *Login*.

Plans

Contents:

- *Basics*
 - *Overview*
 - *Interface*
 - *How To*
-

A **plan** enables the execution of a sequence of discrete tasks in the Designer Cloud powered by Trifacta® platform . These tasks can be flow jobs, workflow jobs, HTTP requests to other platforms, and much more. The sequence of tasks can be gated based on success or failure criteria and can be restarted if needed where the plan failed.

Basics

To get started, see *Plan Basics*.

Overview

Plans provide a mechanism for orchestrating activities within the platform. For more information, see *Overview of Orchestration*.

Interface

For more information on how to create and manage plans, see *Plans Page*.

How To

For how to information on how to create and manage your plans, see *Plan Tasks*.

Plan Basics

Contents:

- *Overview*
 - *Example*
 - *Create a Plan*
 - *Prerequisites*
 - *Steps*
 - *Schedule a Plan*
 - *How To*
-

A **plan** enables the execution of a sequence of discrete tasks in the Designer Cloud powered by Trifacta® platform . These tasks can be flow jobs, workflow jobs, HTTP requests to other platforms, and much more. The sequence of tasks can be gated based on success or failure criteria and can be restarted if needed where the plan failed.

Overview

In many scenarios, data pipelines have dependencies in them. Data sourced from one dataset must be cleaned and delivered for use in another data pipeline. When a pipeline is built, it needs to be scheduled for periodic execution. Ideally, when the execution completes, downstream stakeholders must be informed that the pipeline has succeeded or failed in execution.

In Designer Cloud powered by Trifacta platform , plans provide the mechanism for building these robust data pipelines within the application, in which sequences of tasks can be executed to deliver more diverse datasets. As a result of the success or failure of these flow executions, the Designer Cloud application can deliver messages to receiving applications, such as Slack.

A **plan** is a sequence of tasks.

- A **task** is a type of execution in the Designer Cloud application .
- A **trigger** is a condition under which a task is executed.

Tip: As one of your plan tasks, you can configure an HTTP message, which allows you to trigger endpoints in the Designer Cloud application itself. So, you can configure your plan executions to trigger a wide range of activities within the product itself.

For more information, see *Overview of Orchestration*.

Example

The following example shows how to create a plan that executes the following tasks:

1. Execute the flow task that performs the initial cleaning of your data. The results data is saved to a known location.
2. Execute the flow task that transforms the cleaned data for downstream uses.
3. Based on the outcome of step 2:
 - a. On success: Send a success message to a Slack channel.
 - b. On failure: Send a failure message to a Slack channel.

In Plan View, this plan might look like the following:

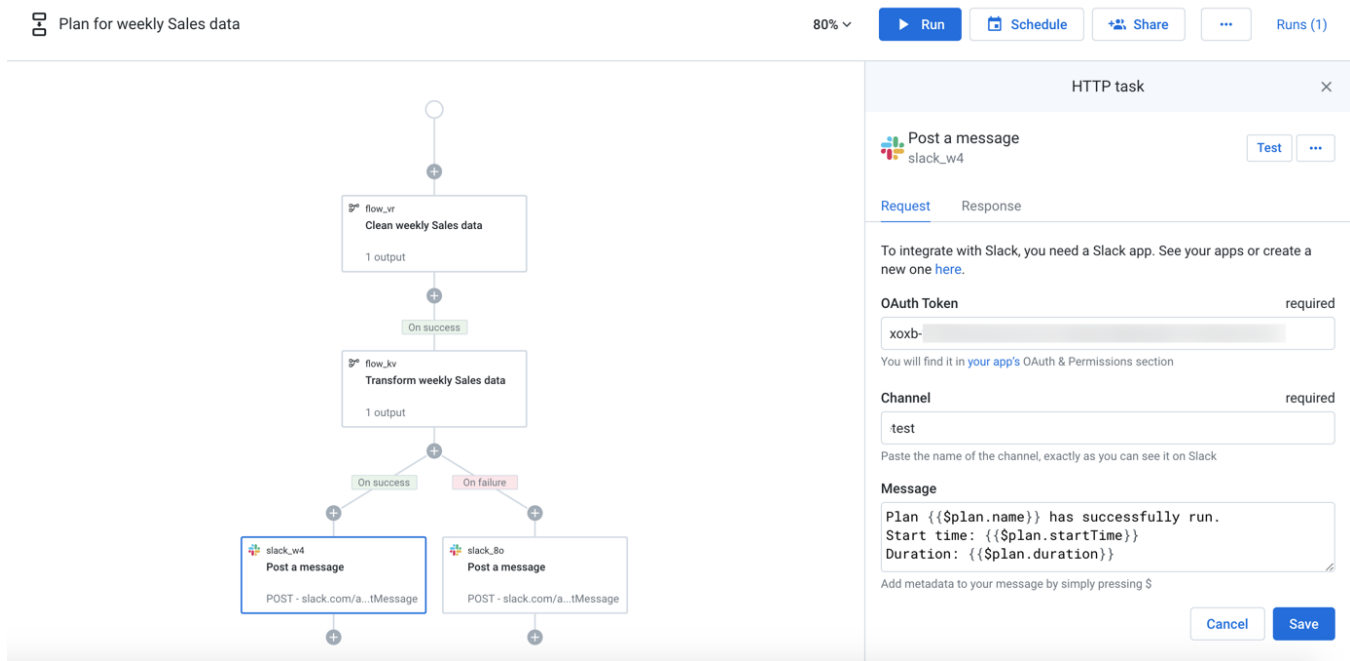


Figure: A plan with a sequence of flow tasks and branching Slack tasks to deliver results messages

After creating the above plan, you can perform test runs.

If the tests are successful, you can schedule the plan for periodic execution.

Create a Plan

Prerequisites

- You must have access to all of the assets that you intend to use it in your plan.
- For each flow in your plan:
 - All of the recipes must have output objects associated with them.
 - Each output object has at least one of the following that is created for it:
 - file-based output
 - table-based output
- Specifying an HTTP task requires knowledge of the target endpoint and the parameters required for the request. HTTP tasks are considered developer-level assets.

Steps

1. From the left navigation bar, click the **Plans** icon.
2. In the Plans page, click **Create**. A new plan with the name `Untitled - x` is created, where `x` is a number.

Tip: Click the `Untitled - x` to enter a plan name and description.

3. In Plan View, click the **Plus** icon in the plan canvas.
4. Select the type of task from the right panel.

a. **Flow task:**

- i. To add a flow task, select **Flow task**. Then, select the output of the flow to generate.

Tip: You are prompted to create an output if there are no outputs associated with the flow.

- ii. To create the flow task, click **Create task**. The task is created and added to the plan.
- iii. To add or remove outputs, click **Add/remove outputs** from the right panel and click **Update task**.

b. **HTTP task:**

- i. To add an HTTP task, select **HTTP task** and enter the required information.

Tip: You can test the HTTP task before you create it. To test for a basic connection and get relevant information, you should only use the GET method.

- ii. To save the task, click **Save**. The configuration is saved and added to the plan.
5. To test the plan, click **Run**.
6. To create a schedule for the plan, click **Schedule** in the top menu bar or click the empty circle in the Plan view page.

Schedule a Plan

You can schedule executions of your plan. See *Schedules*.

How To

For more information on how to create and manage your plans, see *Plan Tasks*.

Overview of Orchestration

Contents:

- *Terms*
- *Task Types*
- *Limitations*
- *Basic Task*
- *Plan Scheduling*
- *Plan Execution*
- *Enable*
- *Logging*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Orchestration is a set of functionality that supports the scheduled execution of task sequences in the Designer Cloud powered by Trifacta® platform . These tasks could be external processes, data transformation jobs, HTTP requests, and more.



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

In the following sections, you can review short summaries of specific features and explore more detailed information on them.

Terms

Term	Description
plan	A plan is a sequence of tasks that are executed on one or more flows to which you have access. To orchestrate tasks, you build a plan. A plan can be scheduled for execution, triggered manually, or invoked via API.
trigger	A task is executed based on a trigger. A trigger is a condition under which a task is executed. In many cases, the trigger for a task is based on the schedule for the plan.
task	A task is a unit of execution in the platform. For example, one type of task is the execution of a flow, which executes all recipes in the flow, as well as the flow's upstream dependencies.
snapshot	A snapshot of the plan is captured, and the plan is executed against this snapshot. For more information on snapshots, see "Plan execution" below.

Task Types

The following types of tasks are available.

Type	Description
flow task	An ad-hoc or scheduled execution of the transformations required to produce one or more selected outputs from a flow.
HTTP task	A request submitted to a third-party server as part of a plan run.
Slack task	Send a message with information about the plan run to a specified Slack channel.
Delete task	Delete files and folders from backend data storage.

Limitations

- You cannot specify parameter overrides to be applied to plans specifically.
 - Plans inherit parameter values from the objects referenced in the plan's tasks.
 - If overrides are applied to flow parameters, those overrides are passed to the plan at the time of flow execution.

Tip: Prior to plan execution, you can specify parameter overrides at the flow level. These values are passed through to the plan for execution. For more information, see *Manage Parameters Dialog for Plans*.

Basic Task

You create a plan and schedule it using the following basic task.

1. Create the plan. A **plan** is the container for definition of the tasks, triggers, and other objects. See *Plans Page*.
2. In Plan View, you specify the objects that are part of your plan. See *Plan View Page*.
 - a. **Schedule:** The schedule defines the set of triggers that queue the plan for execution.
 - i. **Trigger:** A trigger defines the schedule and frequency at which the plan is executed. A plan can have multiple triggers (e.g. monthly versus weekly executions).
 - b. **Task(s):** Next, you specify the tasks that are executed in order.
 - i. **Flow task:** A flow task includes the specification of the flow to run and the outputs from the flow to generate.

NOTE: You can select the outputs from the recipe that you wish to generate. You do not need to generate all outputs.

NOTE: When a flow task is executed, the execution plan works back from the selected outputs to execute all of the recipes required to generate the output, including the upstream dependencies of those recipes.

See *Plan View for Flow Tasks*.

- ii. **HTTP task:** An HTTP task is a request issued when it is triggered from the application to a target endpoint. This request supports a variety of API methods. See *Plan View for HTTP Tasks*.
 - iii. **Slack task:** A Slack task is a message between the Designer Cloud powered by Trifacta platform and a specified Slack channel that is triggered within the plan. See *Plan View for Slack Tasks*.
 - iv. **Delete task:** A Delete task deletes specific files or folders from backend storage. See *Plan View for Delete Tasks*.
 - v. Continue building tasks in a sequence.
3. As needed, you can apply override values to any flow parameters that are included in the tasks of your recipe. These overrides are applied during a plan run. For more information, see *Manage Parameters Dialog for Plans*.
 4. To test:
 - a. Click **Run now**.
 - b. To track progress, click the Runs link.
 - c. In the Run Details page, you can track the progress.
 - d. The first task is executed and completes, before the second task is started.
 - e. Individual tasks are executed as separate jobs, which you can track through the Job History page. See *Job History Page*.
 - f. When the plan has completed, you can verify the results through the Job details page. See *Job Details Page*.

5. If you are satisfied with the plan definition and your test run, the plan will execute according to the scheduled trigger.

Plan Scheduling

Through the Plan View page, you can configure the scheduled executions of the plan. Plan schedules are defined using triggers.

- These schedules are independent of schedules for individual flows.
- You cannot create schedules for individual tasks.

Plan Execution

When a plan is triggered for execution, a **snapshot** of the plan is taken. This snapshot is used to execute the plan. Tasks are executed in the sequence listed in Plan View.

Important notes:

NOTE: Any subsequent changes to the flows, datasets, recipes, and outputs referenced in the plan's tasks can affect subsequent executions of the plan. For example, subsequent removal of a dataset in a flow referenced in a task can cause the task to fail to execute properly.

At the flow level, you can define webhooks and email notifications that are triggered based on the successful generation of outputs. When you execute a plan containing an output with one of these messages, the message is triggered and delivered to stakeholders.

NOTE: Webhook messages and email notification cannot be directly triggered based on a plan's execution. However, you can create HTTP-based tasks to send messages based on a plan task's execution.

Tip: When a flow email notification is triggered through a plan, the internal identifier for the plan is included in the email.

See "Webhooks" and "Email notifications" above.

Enable

Enable the following setting:

Plans feature

Plan sharing, import, and export must also be enabled. For more information, see *Workspace Settings Page*.

The following flags must be enabled for the orchestration service to correctly function.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following settings. Verify that they are set to `true`:

```
"webapp.orchestrationWorkers.enabled": true,
"orchestration-service.enabled": true,
"orchestration-service.autoRestart": true,
```

3. You can choose to enable the following task types:

Task type	Setting	Description
HTTP task	<code>feature.orchestration.httpTasks.enabled</code>	When <code>true</code> , you can configure plan tasks to deliver a REST request over HTTP or HTTPS to a specified endpoint, including endpoints in the Designer Cloud powered by Trifacta platform .
Slack task	<code>feature.orchestration.slackTask.enabled</code>	When <code>true</code> , you can configure plan tasks to deliver messages to a specified Slack channel.
Delete task	<code>feature.orchestration.deleteFileTask.enabled</code>	When <code>true</code> , you can configure plan tasks to delete files or folders from backend data storage.
	<code>feature.orchestration.deleteFileTask.maxFiles</code>	<p>By default, the maximum number of files that can be matched for deletion is 100 . You can modify this value if needed.</p> <div> <p>NOTE: This setting is intended as a safety measure to prevent runaway deletion of a large number of files. Modify this value only if necessary.</p> </div>
Email task	<code>feature.orchestration.emailTask.enabled</code>	This feature is not yet available.
Flow task	This feature is automatically enabled when Plans feature is enabled. See above.	These tasks execute a specific output on a selected flow.

4. Save your changes and restart the platform.

Logging

For more information on debugging plans, see *Diagnose Failed Plan Runs*.

Logging information on plan execution is captured in the `orchestration-service.log`. This log file can be downloaded as part of a Support Bundle. For more information, see *Support Bundle Contents*.

You can configure aspects of how this log captures service events. For more information, see *Configure Logging for Services*.

Plan View Page

Contents:

- *Top Bar*
 - *Plan context menu*
- *Task*
- *Task Execution Rule*
- *Parallel Tasks*
- *View for Triggers*
- *View for Tasks*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

In Plan View, you design your plan, which includes the building and sequencing of tasks and the triggers that execute your sequence of tasks.

NOTE: Access to this page in the application and privileges on its related objects is governed by roles in your workspace. For more information, please contact your workspace administrator.

Random_Data 100% [Run](#) [Schedule](#) [Share](#) [...](#) [Runs \(1\)](#)

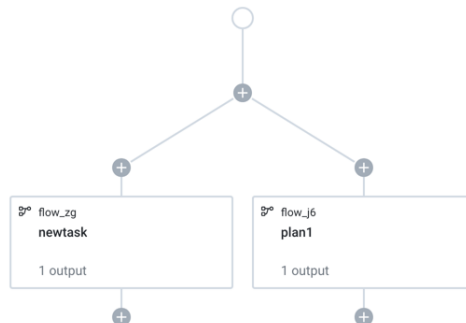


Figure: Plan View Page

The main panel is the plan canvas, where you build your plans.

On the right side is the context panel. Depending on what you select in the plan canvas, a different set of context options is displayed.

Top Bar

Tip: To rename the plan, click the plan name in the top bar.

Zoom Options:

You can zoom the plan canvas to display areas of interest in the plan graph.

The zoom control options are available at the right side of the canvas. The following are the available zoom options:

Tip: You can use the keyboard shortcuts listed in the zoom options menu to make quick adjustments to the zoom level.

- **Zoom in:** Zoom in 10% on the canvas to focus on greater detail.
- **Zoom out:** Zoom out 10% from the canvas to see more of it.
- **Zoom to fit:** Change the zoom level to fit all of the objects of your plan onto the screen.
- **25%, 50%, or 100%:** Change the zoom level to one of the preset levels.

NOTE: By default, the plan view page always opens in Zoom to fit option and it does not remember the previous zoom and position.

Other options:

- **Run:** Run the plan. You can track progress of your plan run. See *Plan Run Details Page*.

Tip: You can apply overrides to parameters through the Parameters tab. See *Plan View for Flow Tasks*.

See *Plan Runs Page*.

- **Schedule:** Create or edit the plan schedule with one or more triggers through the right context panel. See "View for Tasks."
- **Share:** Share the plan with other users. See *Share a Plan*.
- **Runs:** The Runs link tracks the current total number of runs that have been queued or executed for this plan. Click this link to track progress on your plan run.

Plan context menu

- **Rename:** Modify the name and description for your plan.
- **Parameters:** You can apply overrides to the recipe parameters for your plan tasks during plan job runs. See *Manage Parameters Dialog for Plans*.
- **Email notifications:** Send email notifications on the plan runs. See *Manage Plan Notifications Dialog*.
- **Export:** Export the plan from the Designer Cloud powered by Trifacta platform . See *Export Plan*.
- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user. See *Transfer Asset Ownership*.
- **Delete:** Delete your plan.

Task

For more information, see *Create a Plan*.

Task Execution Rule

You can gate the execution of a task based on the completion status of its previous task. Click the line connecting the two tasks and select one of the following options:

- **On success:** Runs if the previous task in the node is successful.
- **On failure:** Runs if the previous task in the node failed.
- **On execution (any status):** Runs the task in the node irrespective of the previous task's status (success or failure).

Parallel Tasks

You can execute one or more tasks in parallel by clicking the plus node icon and selecting one of the following options:

- **Add a parallel node:** Adds a parallel node to the existing node.
- **Add a node:** Adds a node at the bottom of the existing node.

NOTE: Nodes added in a sequence are separated by a plus node. A node added in parallel has two plus nodes separating it from the parent node; one for adding nodes in parallel and other for adding a node in sequence.

View for Triggers

For more information, see *Plan View for Triggers*.

View for Tasks

When you create or select a task, you can modify its settings through the context panel on the right.

Tip: To rename the task, click the task name in the context panel.

Task context menu options:

Some options may not be available for specific task types.

- **View flow:** This option opens the flow.
- **Edit name:** Modify the name of the task.
- **Delete:** Delete the task from your plan.

Task types

- **Flow task:** Generate all of the defined output objects for a flow. See *Plan View for Flow Tasks*.
- **HTTP task:** Execute a task over HTTP protocol. See *Plan View for HTTP Tasks*.
- **Slack task:** Send a message from the Designer Cloud application to a specified Slack channel. See *Plan View for Slack Tasks*.
- **Delete task:** Delete files or folders from backend data storage as a task. See *Plan View for Delete Tasks*.

To cancel a plan that is currently running, please do the following:

- In Plan View, click the **Runs (x)** link.
- In the Run Details page, click **Cancel plan run**.

Plan Run Details Page

You can review the details of individual executions of your plan. In Plan View, click the **Runs** link in the upper-right corner.

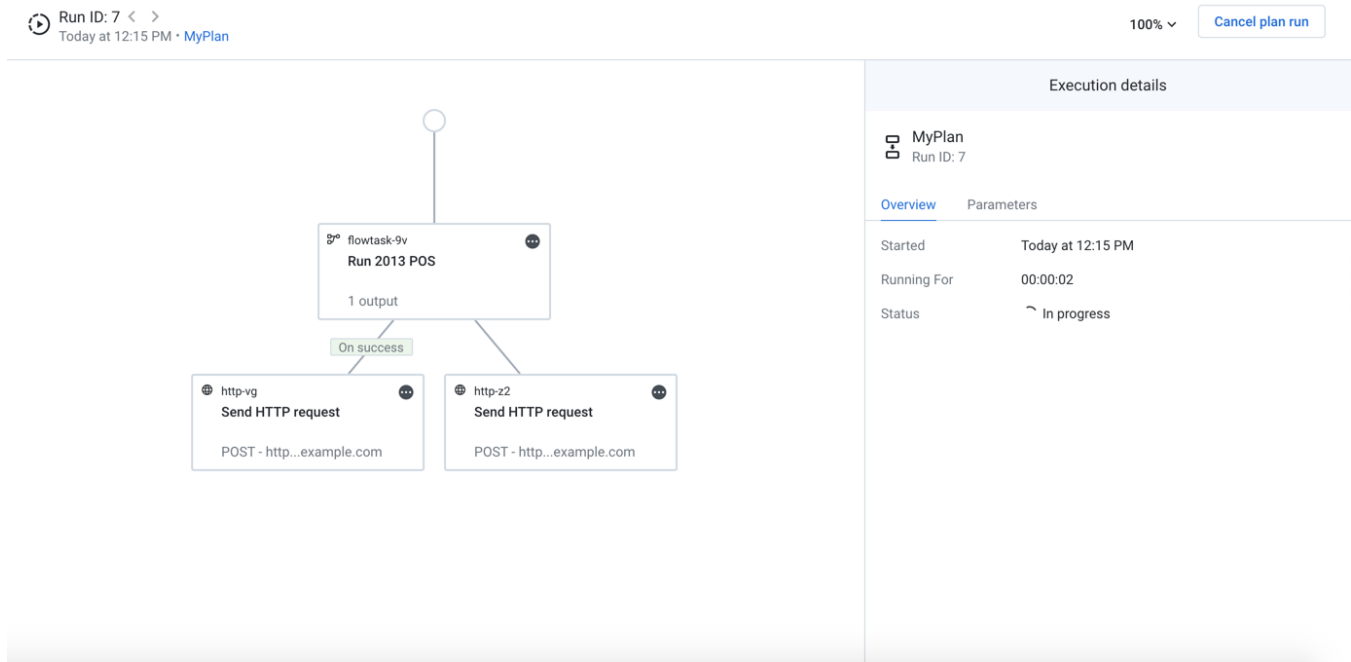


Figure: Run details

The latest plan run is displayed. You can review the progress of individual tasks throughout the plan run.

Tip: You can click the caret next to the Run ID to scroll back and forth to view multiple runs of the plan.

NOTE: When a plan run begins, a snapshot is taken of the plan. Subsequent changes to the underlying assets could impact the outcome of the asset tasks when they are later executed during the plan run.

- You can select individual triggers and tasks to review details of the plan run for that object in the context panel.
- To see other runs for the plan, use the angle brackets next to the timestamp at the top of the screen.

To cancel a plan that is currently running, please do the following:

- In Plan View, click the **Runs (x)** link.
- In the Run Details page, click **Cancel plan run**.

You can track all of the runs across all of your plans. See *Plan Runs Page*.

Retry Failed Plan Runs

To retry a plan run where one or more tasks failed, click **Retry from failed**. Plan tasks that were completed successfully are skipped, and the new plan run begins with the task or tasks that failed.

Following are some scenarios where you can restart the plan runs:

- When a plan run fails at a specific task, the failed task is retried and the downstream tasks are executed.
- When a plan run fails in multiple tasks, all the failed tasks and their respective downstream tasks are retried.
- If two tasks in the same path fail, the first one acts as the starting point.







In the Retry plan run dialog, you can:

- View the tasks that failed as starting points for the retry.
- Reuse or override any parameter values used in the original plan run.

To run the entire plan again, click **Run plan** in Plan View page.

Task Execution

The following icons indicate the results of the execution of a task:

Icon	Task Status
	Task successful
	Task failed
	Task skipped. Task was not executed due to unmet conditions.
	Task canceled by user
	Task in progress
	Task pending

Share Plan Dialog

You can manage access to a plan for other users through the Share Plan dialog. From the context menu of the Plans page, select **Share**.

Tip: If groups have been enabled in your instance of the Designer Cloud powered by Trifacta platform , you can share flows and connections to LDAP groups. For more information, see *Configure Users and Groups*.

Permissions

You can grant permissions to other users to access the plan. When a user is given access to a plan, that user is considered a **collaborator** on the plan and has a smaller set of permissions than the **owner** of the plan. Users must have at least the Viewer role viewer permissions for plans.

For more information, see *Roles Page*.

NOTE: If the user does not have access to the underlying assets of the plan, the plan can still be shared and accessed, but the user cannot edit those assets and run the plans.

Actions

Through this Share Plan dialog, you can invite one or more collaborators to the plan, so that you may work together on the same objects.

- **Add:** Add users or email addresses of users with whom you would like to share the plan.
 - To add users as collaborators, start typing the name of a user with whom you would like to share the plan.
 - Select the user.
 - Specify the privilege level of the user to whom you are sharing. For more information on sharing privileges, see *Overview of Sharing*.
 - Repeat this process to add multiple users.
- **Save:** Click **Save**.
- **Delete:** Select the user or email address in the search by name or email field and click delete using your keyboard.
- **Cancel:** To cancel sharing, click **Cancel**.

Each selected user now can access the plans through the plans page. See *Plans Page*.

Manage Plan Notifications Dialog

This section provides an overview of sending email notifications to plan owners and collaborators based on the results of plan runs.

When email notifications are enabled, plan owners and collaborators can specify the list of email recipients, based on the status of execution of plans. From the context menu of the Plans page, select **Email notifications**.

- Users who receive notifications for specific plans are considered plan watchers.
 - By default, all collaborators receive notifications about plan run failures.
 - If plan collaborators have only view permissions, they may not be able to edit the recipients.
 - You cannot enable or disable email notifications at the plan level. Workspace administrators can enable or disable email notifications for plans.
- For more information, see *Workspace Settings Page*.

NOTE: This feature requires access to an SMTP server to send emails. For more information, see *Enable SMTP Email Server Integration*.

Tip: Email recipients can remove themselves from receiving notifications on plan runs using a link at the bottom of the email.

Email notifications

Send email notifications when this plan runs

Add recipients

Add

Recipient	Notify
<div><div>TU</div><div>Test User 4147080222</div><div>4147080222@trifacta.com</div></div>	<div>On failure</div> <div></div> <div></div>
<div><div>A</div><div>Administrator</div><div>admin@trifacta.local</div></div>	<div>On failure</div> <div></div> <div></div>

Done

Figure: Manage Plan Notifications

NOTE: You can enable or disable the **Send email notifications when this plan runs** option to enable or disable plan email notifications. If this option is disabled, the below options are not available.

For more information, see *Email Notifications Page*.

In the Email notifications dialog, plan owner and collaborators can add the email addresses to stakeholders to receive notifications based on the plan run status:

- **On success:** Emails are generated if the plan run succeeded.
- **On failure:** Emails are generated if the plan run failed.
- **On execution (any status):** Emails are generated whether the plan run succeeded or failure.

Actions:


- **Add:** Add email addresses in the **Add recipients** field.

Tip: Users can send plan run emails to any valid email address or email alias, even if the user(s) do not have an account in the Designer Cloud powered by Trifacta® platform .

- **Delete:** To remove a user, click the Trash icon next to the email address.
- **Save:** Click **Done**.


Manage Parameters Dialog for Plans

You can specify parameter overrides within your plans. Flow parameters are available for modification at the plan level, and their values or overrides are applied during plan runs.

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Manage parameters

×

Parameter	From	Created in	Value
< > region	Run Datasets - WindowsFunctions Flow flowtask-32	 Datasets - Window...	<div>02</div> <div>CancelSave</div>

Close

Figure: Manage Parameters Dialog for Plans

In the Manage Parameters dialog, you can review the parameters that are defined for the flows that are part of your plan.

NOTE: You cannot create parameters through Plan View. You must create the parameters within the flows that are part of your plan. After they are created, they are available for overrides through Plan View.

NOTE: In a plan, parameters and their overrides apply only to the flow task from where they are sourced. They do not apply to other flow tasks. They do not apply back to the source flows.

Tip: To pass a flow parameter value from one recipe to another, you can insert the parameter value in a column in a recipe, export the results as a reference dataset, and then ingest that reference into another flow.

Columns:

- **Parameter:** The name of the parameter
- **From:** The source flow in your plan for the parameter
- **Created in:** The object in the source flow where the parameter is defined
- **Value:** The current value for the parameter

Tip: To override the inherited value for the parameter, click the Pencil icon. Then, click **Save**. Whenever the flow is executed as part of this current plan, the override value is used to replace the parameter value and any override that are defined within the flow.

Plan View for Flow Tasks

Contents:

- *Sources tab*
 - *Outputs tab*
 - *Parameters tab*
 - *Troubleshooting*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When you create a flow task in Plan View, you first search for the flow that you wish to run. Then, you specify the task to execute on the flow in the right context panel.

When a task is executed, the recipes in the flow that have been selected in the task are executed, and if successful, their selected outputs are generated.

NOTE: In a flow, all recipes that you wish to have executed by the corresponding task must have a defined output object. For each output object, you must create at least one write settings or publication object. During plan runs, these objects are not validated, and missing outputs are ignored.

Run flow

×

Q Search flows...

2013 POS

POS-r01 – 2.txt

Run-Job-Publishing-2020-04-16T20:28:40.141Z

allTypes

Run-Job-Publishing-2020-04-16T20:02:41.587Z

allTypes

Untitled Flow

No outputs

flowy

job – 2.log

Test-Notification

carriers_recipe, Flights_recipe

Test-Notification


carriers_recipe, Flights_recipe

Figure: Flow task - select flow

- Enter a search string in the search box.
- When you locate the flow to execute, select it.

Select the outputs that you wish to have generated for the flow.

< Run flow ×

 Run-Job-Publishing-2020-04-16T20:02:41.587Z

Choose outputs to run

☒ allTypes

Cancel Create task

Figure: Flow task - select outputs

Webhooks: If webhooks are configured for the underlying flow, you can optionally disable execution of the webhooks that are defined in the flow when the flow task in the plan is executed. Click the icon to enable or disable webhook execution.

To save the task, click **Create task**.

The saved task is displayed in the context panel.


Tip: To rename the task, click the task name.

Sources tab

Review the datasources for the task, including a full path to the source location.

Run flow

×


 Run Run-Job-Publishing-2020-04-16T20:02:41.587Z

Flow: Run-Job-Publishing-2020-04-16T20:02:41.587Z

...

Sources

Outputs

 allTypes.csv

hdfs://

trifacta/uploads/7/efc8632e-0400-4382-ba3f-88d4d2c6ac8a/allTypes.csv


Figure: Flow task - Sources tab

Outputs tab

Review the outputs that you have selected for the task to generate.

Run flow

×




Run Run-Job-Publishing-2020-04-16T20:02:41.587Z

Flow: Run-Job-Publishing-2020-04-16T20:02:41.587Z

...

Sources

Outputs



allTypes

Profile results

1 publishing action - [Show](#)

[Add/remove outputs](#)

Figure: Flow task - Outputs tab

- To review the publications configured for the output, click **Show**.
- To change the outputs generated by the task, click **Add/remove outputs**.


Parameters tab

If the flow from which the task was created contains parameters, you can review those parameters and apply overrides as necessary.

NOTE: Parameter overrides applied to a plan affect only plan execution. These overrides do not apply to any independent job executions of the underlying flows.

Flow task

×



Run 2013 POS

flowtask-9v • [2013 POS](#)

...

Sources

Outputs

Parameters

Parameter	Value	Manage
< > region	02	

Figure: Flow task - Parameters tab

To apply overrides:

1. Click **Manage**.
2. In the Manage Parameters dialog, mouse over the Value entry for the desired parameter. Click the Edit icon.

Tip: You can use the Search box to locate parameters by name.


3. Enter a new value and click **Save**.

Whenever the flow task is executed, this override value is applied to the execution of the flow task. For more information, see *Manage Parameters Dialog for Plans*.

Troubleshooting

Flow tasks are essentially standard transformation jobs with additional metadata on them. When a flow task fails to execute for any reason, it's likely that the source of the problem resides in the flow definition, its objects, and its connections. For more information, see *Diagnose Failed Jobs*.

Plan View for HTTP Tasks


 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

In Plan View, you can create HTTP tasks to send request to endpoints before or after the execution of other tasks. These tasks are specified in the right context panel.

<

HTTP task

X

 Send HTTP Request

Test

...

Method

GET

▼

URL

required

https://example.com/v4/connections

https://example.com/endpoint

Headers ?

Add

Authorization

<my_authorization_key>

Remove

Key

Value

Remove

Secret Key

Secret Key

☒ Validate SSL certificate

Retry

3

times

Cancel

Save

Figure: HTTP task


Fields:

--	--

Field	Description
Method	Select the HTTP method to use to deliver the message. The appropriate method depends on the receiving application. Most use cases require the POST method.
URL	URL where the HTTP request is received by the other application.
Headers	<p>Insert HTTP content headers as key-value pairs. For example, if your body is in JSON format, you should include the following header:</p> <pre>key: Content-Type value: application/json</pre> <p>NOTE: You may be required to submit an authentication token as the value for the <code>Authorization</code> key.</p>
Body	<p>(POST , PUT , or PATCH methods only) The body of the request submitted to the receiving application. Request body is structured as follows:</p> <pre>{"text": "My text message to the receiving application."}</pre> <p>Tip: As part of the request body or header fields, you can insert metadata references to the plan definition, current run, tasks already executed in the run, and, column data and data sources. For more information on the available metadata, see <i>Plan Metadata References</i>.</p> <p>For examples of requests including metadata examples, see <i>Create HTTP Task</i>.</p>
Secret Key	<p>(Optional) A secret key can be used to verify the request payload. This secret value must be inserted in this location, and it must be included as part of the code used to process the requests in the receiving application. Insert the secret value here as a string without quotes.</p> <p>For more information on how this secret key is used to generate a signature, See <i>Create HTTP Task</i>.</p>
Validate SSL Certificate	<p>When set to <code>true</code>, HTTPS (SSL) communications are verified to be using a valid certificate before transmission.</p> <p>NOTE: If you must send a request to an endpoint that has an expired/invalid certificate, you must disable SSL verification.</p>
Retry	<p>If the returned status code is outside of the 200-299 range, then the HTTP task is considered to have failed. When this option is enabled, the request is retried.</p> <p>If the request fails, this value defined the number of times that the request should be retried. If this number of retries is reached without success, the task fails.</p>

For more information, see *Create HTTP Task*.

Plan View for Slack Tasks

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.


In Plan View, you can create tasks to send messages to a Slack channel. These tasks are specified in the right context panel.

Tip: Slack tasks are a specialized form of HTTP tasks.

<

Slack task

×

 Post a message

Test

...

Request

Response

To integrate with Slack, you need a Slack app. See your apps or create a new one [here](#).

OAuth Token

required

xoxb

You will find it in [your app's](#) OAuth & Permissions section

Channel

required

testing

Paste the name of the channel, exactly as you can see it on Slack

Message

Plan {{plan.name}} is running.
Start time: {{plan.startTime}}

Add metadata to your message by simply pressing \$

Cancel

Save

Figure: Slack task


Fields:

Field	Description
-------	-------------

OAuth Token	The OAuth token to use for posting the message.
Channel	<p>Paste one of the following values from the Slack workspace for where to post the message:</p> <ul style="list-style-type: none"> • Channel Name: Name of the channel as it appears in Slack. • Channel ID: This value is available in the Settings page for the channel. • Member ID: You can post the message to a specific user instead of posting to a channel. <div> <p>Tip: A user's member ID can be found in the user's Profile page in Slack.</p> </div>
Message	<p>The message to post.</p> <div> <p>Tip: Messages can include metadata information about the tasks in the current plan run. For more information, see <i>Plan Metadata References</i>.</p> </div>

For more information, see *Create Slack Task*.

Plan View for Delete Tasks

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.


A Delete task can be created in Plan View to delete existing files or folders from backend storage. These tasks are specified in the right context panel.

Tip: Delete tasks are useful for removing files that were generated as part of plan's execution but are not needed afterward.

<

Delete task

×

 Delete


...

Path

@trifacta.com/jobrun/POS-r01.json

Browse

Files to delete (1)

 @trifacta.com/jobrun/POS-r01.json

Cancel

Save

Figure: Delete task

Fields:

Field	Description
Connection	<p>If you have access to multiple file-based connections, you can select the connection from the Connection drop-down.</p> <p>If this drop-down is not present, you automatically connect through the base storage layer for your environment.</p>
Path	<p>Specify the location where you wish to remove files. To navigate the storage environment, click Browse.</p> <div><p>Tip: You can paste in the Path textbox values that you have copied.</p></div> <div><p>Tip: You can insert plan metadata references in the path for tasks that have previously been executed in the plan. Enter \$ to begin exploring available references.</p></div> <p>You can select entire folders. These folders and files must exist at the time of creating the Delete task.</p> <div><p>NOTE: As a safety measure, you are not permitted to delete more than 100 files in a single task.</p></div>

For more information, see *Create Delete Task*.

Plan View for Triggers



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

To create a trigger for a plan, you specify the time and frequency that the plan is to be executed. You can also apply overrides to any inherited parameters.

NOTE: Plan owners can create triggers for their plans. Other users must have the appropriate privileges to create triggers for the plan.

<
Trigger
×

Name

Plan-2021 POS schedule

Timezone

America/Los_Angeles

Frequency

Monthly

[Add another trigger](#)

On the

1

day(s) of the month

at

12:00

AM

Scheduled plans will run as the plan owner. [Learn more](#)

Parameter overrides
Set parameter values for this schedule's executions

<> region flow_7j	4
<> year flow_7j	2021

Delete

Cancel

Save

Figure: Create plan trigger

- **Timezone:** Select the timezone for the time that you are specifying in the trigger's schedule.
 - To use UTC time zone, select `UTC` in the drop-down.
 - For a list of supported timezones, see *Supported Time Zone Values*.
- **Frequency:** Specify how frequently the plan is triggered:
 - **On:** You can specify multiple entries to trigger the plan more frequently.
 - **cron:** Set the schedule according to cron syntax.
 - Time zone settings set in the drop-down are used with the cron schedule.

- For more information, see *cron Schedule Syntax Reference*.
- **Parameter overrides:** See below.
- To add another trigger, click **Add another trigger**.
- To save your changes, click **Save**.

Parameter Overrides

As needed, you can specify overrides to parameters inherited from the tasks in a plan. Specified values are applied only when the plan run is executed based on this trigger.

NOTE: In a trigger, displayed parameter values may be inherited from the plan that is being triggered. To ensure that the proper value is used, you should set a specific value for the override in the trigger. This is a known issue.

To apply a parameter override:

- Locate the parameter to override.
- Verify that the parameter is sourced from the correct object.
- Click **Override**.
- Enter the value to apply.
- Repeat as needed for other parameter overrides.
- Click **Save**.

For more information, see *Overview of Parameterization*.

Enable or Disable Trigger

After a trigger has been saved, you can enable or disable it when the trigger node in the plan is selected.

Trigger

×

Schedule

Enable ☒ Edit

Weekly: 12:10 AM

On Sunday and Wednesday

Monthly: 12:00 AM

On the 1st

Figure: Saved trigger

- To disable the trigger, click the slider.

NOTE: If you disable a trigger, no new scheduled executions of the tasks in the plan occur. You can still manually trigger plan runs.

- To make changes to the trigger, click **Edit**.

Plans Page



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

The Plans page lets you create, review, and manage your plans. A **plan** is a sequence of tasks and the triggers that execute them.

NOTE: Access to the Plans page in the application and privileges on plans is governed by roles in your workspace. For more information, please contact your workspace administrator.

- To create a new plan, click **Create....** To rename the new plan, click the `Untitled` value at the top of the page.
 - The maximum number of plan tasks may vary depending on the plan that you have licensed. For more information, please contact your Trifacta representative.
- Workspace admins can access all plans in the workspace.

Name	Owner	Last updated	Last run
Shared with 3 collaborators	A	09/23/2021	09/23/2021
	SO	09/18/2021	09/20/2021
	SO	09/05/2021	Last Sunday at 12:30 PM
test_plan	A	08/14/2021	-
Untitled plan	A	08/14/2021	-

Figure: Plans Page

Columns:

- **Name:** The name of the plan.

You can hover over the shared icon link next to the plan to view the name of shared users (up to three shared users) and the total number of shared users. Also, when you click the shared icon link, the share dialog is displayed.

- Click the plan name to review it. See *Plan View Page*.

- **Owner:** Owner of the plan.
- **Last Updated:** Timestamp for the last modified time.
- **Last Run:** Timestamp for when the plan was last executed.
 - The displayed icon indicates whether the plan executed successfully or not.
 - Click the link to review details of the run.

Actions:

- **Create:** From the Create menu, choose to create a plan.
 - Enter a name and description for your plan. Click **Create**.
- **Import:** From the context menu, select **Import** to import a plan into this instance. See *Import Plan*.

Tip: You can import multiple plans (ZIP files) through the file browser or through drag-and-drop. Press **CTRL/COMMAND + click** or **SHIFT + click** to select multiple files for import.

- **Search:** To search plan names, enter a string in the search bar. Results are highlighted immediately in the Plans page.
- **Sort:** Some column headers can be selected to sort the display by the column's entries.

Plan options:

The following options are available on the right side of a plan's entry:

- **Share:** Share the plan with other users. See *Share a Plan*.
- **Rename:** Change the name and description of the plan.
- **Email notifications:** Send email notifications on the plan runs. See *Manage Plan Notifications Dialog*.
- **Export:** Export the plan from the Designer Cloud powered by Trifacta platform . See *Export Plan*.
- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user. See *Transfer Asset Ownership*.
- **Delete:** Delete the plan.

Deleting a plan removes all objects contained in the plan. Flows referenced in the tasks of the plan are not touched.

Plan Tasks

Build plans to handle orchestration of tasks within the platform and communication with external systems.

Create a Plan

Contents:

- *Before You Begin*
 - *Create Plan*
 - *Create Task*
 - *Task Types*
 - *Apply Parameter Overrides*
 - *Create Plan Branches*
 - *Add Task Execution Rule*
 - *Add Parallel Task*
 - *Example - Success or Failure Tasks in a Plan*
 - *Test Plan*
 - *Monitor Plan Runs*
 - *Add Plan Schedule*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

A **plan** is a sequence of tasks that are executed manually or based on a schedule. Plans can be used to automate the execution of multiple related tasks in the Designer Cloud powered by Trifacta® platform .

- When a plan is triggered:
 - A snapshot of the objects in the plan is capture. This snapshot defines the set of tasks that are executed as part of a plan run.

NOTE: A snapshot does not capture the assets underlying the tasks. After a snapshot is taken, subsequent changes to the underlying assets could impact the outcome of the tasks when they are later executed during the plan run.

- The set of tasks in the plan are triggered in the order listed in the plan.
 - All of the dependencies for any task are also executed.
 - If one task fails to execute, the other tasks are not executed.
- For more information on plans, see *Overview of Orchestration*.

Before You Begin

Before you begin, please verify the following:

- You have access to all of the assets that you wish to use in your plan.
- For each flow in your plan:
 - All of the recipes whose results you wish to generate have output objects associated with them.
 - Each output object has at least one of the following that has been created for it:
 - file-based output
 - table-based output

NOTE: In a flow, all recipes that you wish to have executed by the corresponding task must have a defined output object. For each output object, you must create at least one write file or table settings definition. During plan runs, these objects are not validated, and missing outputs are ignored.

Create Plan

To begin, you must create a plan object.

Steps:

1. From the left nav bar, click the Plans icon.
2. The Plans page is displayed.
3. In the Plans page, click **Create**. A new plan with the name `Untitled - x` is created, where `x` is a number.
4. Click the `Untitled - x` to enter a plan name and description.
5. Your plan is saved and displayed in Plan View.

In Plan View, you create the objects that are part of your plan. These include:

- **Plan Schedule:** A schedule is composed of one or more triggers that determine when the plan is executed.
 - **Trigger:** Scheduling object that determines the conditions under which the plan is executed.
 - A schedule can contain one or more triggers.
- **Task:** An action that is executed when triggered.
 - You can build a sequence of one or more tasks in your plan.

Create Task

Steps:

1. Identify the tasks that you wish to execute. See "Task Types" below.

NOTE: You must have access to any assets that you wish to execute as tasks.

2. Add a task.
 - a. In Plan View, click the Plus icon at the bottom of your plan.
 - b. Specify the task to execute.
3. Repeat the previous step to add additional tasks as needed.

Tip: You can insert tasks between other tasks. Use the Plus icon between two plan objects.

4. To test your plan, click **Run now**. The plan is immediately queued for execution.
5. Edit the plan and repeat the above steps until the plan is ready for production runs.

Tip: While a plan is in development, you may wish to disable its schedule, which prevents execution according to the schedule. You can still run test executions using the Run Now button.

6. Create the schedule for the plan.
 - a. In the context menu for the plan, select **Schedule**.
 - b. Specify one or more triggers for the schedule. When a trigger occurs, the plan is queued for execution.
7. When ready, the plan runs at the time scheduled in the trigger.

Task Types

Based on the schedule's triggers, you can define a sequence of one or more tasks that are executed.

- To add a new task, click the + icon below the trigger. Select the type of task in the right panel.
- To insert a task between two other objects, click the + icon between them.

Add run flow task

A **flow task** executes the recipes that produce the output objects of the flow.

Steps:

1. After you select the flow task type, use the Search bar or browse to select the flow that you wish to add as the task.
2. Select the output or outputs that you wish to generate from the selected flow.
3. Click **Create task**.
4. The task is created and added to the plan.

For more information, see *Plan View for Flow Tasks*.

Add HTTP task

An **HTTP task** is a request sent using HTTP protocol to a target URL, which could be a REST API endpoint.

NOTE: Specifying an HTTP request requires knowledge of the target endpoint and the parameters required for the request. HTTP tasks are considered developer-level objects.

Steps:

1. After you select the HTTP task type, you can specify the task in the context panel.
2. Specify the fields of the request.

Tip: If possible, you should test the HTTP task before you create it. To test for basic connection, you should use the GET method, which just returns relevant information. Some other methods are potentially destructive.

3. Click **Save**.
4. The task is created and added to the plan.

For more information, see *Create HTTP Task*.

Add Slack task

A **Slack task** is a message submitted from the Designer Cloud application to a specified Slack channel.

Steps:

1. After you select the Slack task type, you can specify the task in the context panel.
2. Specify the fields of the request.
3. Click **Save**.
4. The task is created and added to the plan.

For more information, see *Create Slack Task*.

Add Delete task

A **Delete task** deletes a specified set of files or folders from backend storage.

Steps:

1. After you select the Delete task type, you can specify the task in the context panel.
2. Specify the path to the file or folder to delete. This path must already exist.

- a. **Location:** If this drop-down is available, select the file-based connection. To explore this connection, click **Browse**.
 - b. Navigate to your preferred destination. Click **Choose**.
3. Click **Save**.
4. The task is created and added to the plan.

For more information, see *Create Delete Task*.

Apply Parameter Overrides

If your plan tasks include flows in which parameters have been defined, you can review and override these parameter values. Overrides are applied when the task is triggered as part of a plan run.

Steps:

1. From the Plan View context menu, select **Parameters**.
2. Review the names, sources, and current values for all of the parameters in your plan.
3. To apply an override, click the Pencil icon and enter a new value. Click **Save**.

Subsequent runs of the plan use this new value as the override for the parameter. For more information, see *Plan View for Flow Tasks*.

Create Plan Branches

In some scenarios, you may need to branch plan execution steps based on the results of a task in the plan. For example, you may need to send separate messages using an HTTP task depending on whether a previous task succeeds or fails in execution. You can create branches in the plan graph by adding task execution rules and parallel nodes, which run based on the success and failure states of your plan runs.

To begin this simple example:

1. Create your first task, which is a flow task in the above example. For more information, see *Add Tasks* above.
2. Complete the following sections.

Add Task Execution Rule

Next, you create the first HTTP task that results from the above task and the execution rule that determines when it runs.

- This task should run based on the successful execution of the flow task.
- A **task execution rule** is a condition that is tested after a flow task has run to determine if the task that is downstream of it is executed as a result. In this case, you create an `On success` rule.

Steps :

1. Click the plus icon below the existing flow task node.
2. Select the HTTP task type and enter information in the required fields. See *Add HTTP Tasks* above.
3. Click the link connecting the created HTTP task node and its previous task node and select `On success`.

The HTTP task is executed only when the flow task has run successfully.

Add Parallel Task

Next, you can create the HTTP task that runs when the flow task fails.

Steps:

1. Click the plus icon below the existing flow task node and select **Add a parallel node**. A parallel node is added to the plan graph. See Example below.
2. Select the HTTP task type and enter information in the required fields. See Add HTTP Tasks.
3. Click the link connecting the new HTTP task and its previous task node and select `On failure`.

The second HTTP task is executed only when the flow task has failed to execute.

Tip: You can use parallel tasks to create separate paths through a plan when there are no dependencies between the paths.

Example - Success or Failure Tasks in a Plan



Figure: Success and Failure tasks

When the flow tasks complete successfully, the `On success` HTTP task sends a message.

When the task fails, the `On failure` HTTP task delivers a different message.

Test Plan

After you have created the tasks of your plan, you can perform a test run of the plan.

Steps:

1. To test, click **Run now**.
2. The plan run is queued for execution.

Monitor Plan Runs

1. in the upper-right corner of Plan View, click the Runs link.
2. In the Plan Runs page, you can track the progress of your plan run.
 - a. The most recently triggered plan run is displayed.
 - b. If you have executed multiple runs, you can use the angle brackets next to the timestamp for the run.
3. For tasks in progress, you can click the task to display information in the context panel.
4. To see the details for the plan run, click the Outputs tab. Then, click **Job details**.

For more information, see *Plan Runs Page*.

Add Plan Schedule

You can add a schedule object to specify the triggers when the plan is to be executed.

NOTE: A plan's schedule cannot be executed until its schedule has been enabled. If a plan has a disabled schedule, you can still execute it via the Run Now button.

Steps:

1. To begin, do one of the following, click **Schedule** from the Plan View context menu.
2. In the right context panel, click **Create schedule**.

3. In the Add Trigger panel, you can specify the triggers when the plan is executed. You can specify one or more triggers:

The screenshot shows the 'Trigger' configuration panel. It includes a title bar with a back arrow, the text 'Trigger', and a close 'X' button. The panel is divided into two main sections for configuring triggers. The first section has a 'Timezone' dropdown menu set to 'America/Los_Angeles', a 'Frequency' dropdown menu set to 'Weekly', and an 'On' dropdown menu set to 'Sunday' with a delete 'X' button. Below this is an 'At' section with a time input field set to '12:15' and an 'AM' dropdown menu. The second section is identical but with the 'On' dropdown menu set to 'Wednesday'. At the bottom right of the panel are 'Cancel' and 'Save' buttons.

Figure: Add trigger(s)

4. For each trigger:
- Timezone: Specify the timezone that applies to the scheduled time. For more information on timezones, see *Supported Time Zone Values*.
 - Frequency: You can specify the frequency of when the schedule is triggered.
 - In each trigger, you can specify multiple On values (e.g. Same time on Sunday and Monday).
 - As needed, you can specify the On value using a modified form of cron job syntax. For more information, see *cron Schedule Syntax Reference*.
5. To add more triggers, click **Add another trigger** and specify it.
- To delete a trigger, click the X next to it.
6. **Parameter overrides:**
- If the assets in your plan contain parameters, you can apply overrides to the parameter values.
 - Overrides provided in this panel are applied only when the trigger is executed.

NOTE: Multiple values are ok for plan parameters, as long as the parameter values do not conflict. If you see a warning icon next to a set of multiple parameter values, then you must fix this conflict in the original asset, or the plan fails to execution.

- You can apply overrides through Plan View, too.

7. To save your schedule, click **Save**.
8. In the context panel, you can make changes to your schedule:
 - a. After saving, the schedule is automatically enabled. To disable the schedule, use the slider bar.

NOTE: A plan cannot be executed if the schedule for it has been disabled.

- b. To make changes to the schedule and its triggers, click **Edit**.

Create Delete Task

Contents:

- *Limitations*
 - *Prerequisites*
 - *Create Task*
 - *Rename Task*
 - *Delete Task*
 - *Plan metadata references*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create plan tasks to delete existing files or folders through connections to which you have access. These tasks are helpful for removing files that were generated as part of intermediate steps in your plan's execution.

- A Delete task is defined as one of the tasks in a plan. For more information, see *Plan View Page*.
- This capability may need to be enabled in your environment. For more information, see *Overview of Orchestration*.

Limitations

- As a safeguard, you are prevented from deleting more than 100 files at a time. The maximum file limit for delete tasks can be modified, if needed. See *Overview of Orchestration*.
- Delete tasks are supported for the following file systems:
 - S3
 - ADLS

Prerequisites

- You must have write access to the connection, bucket, and folder where you wish to delete files.
 - You must have access to the connections through which a Delete task removes files or folders.
- You must have write permissions to:
 - Any connection through which you are removing files or folders.
 - Any bucket, folder, or file accessed through that connection.

NOTE: If you select to delete a bucket, the contents of the bucket are removed, but the bucket object remains.

Create Task

1. Open your plan in Plan View. In your sequence of tasks. Click a Plus sign icon to create a new task.

2. In the right panel, select **Delete task**. The Delete task panel is displayed.

< Delete task X

Delete ...

Path

@trifacta.com/jobrun/POS-r01.json Browse

Files to delete (1)

@trifacta.com/jobrun/POS-r01.json

Cancel Save

Figure: Delete task

3. Set the required parameters. For more information, see *Plan View for Delete Tasks*.
4. To add the task to the flow, click **Save**.

Rename Task

To rename the task, click **More menu > Edit** in the right panel.

Tip: Good naming may include the target platform endpoint and method, as well as the purposes of the task in your plan.

Delete Task

To delete the task, click **More menu > Delete**. Confirm that you wish to delete the task.

This step cannot be undone.

Plan metadata references

Within the message of your other tasks, you can reference metadata about the plan, including this task. For more information, see *Plan Metadata References*.

Create HTTP Task

Contents:

- *Limitations*
- *Prerequisites*
 - *Requirements for receiving application*
- *Create Task*
- *Rename Task*
- *Delete Task*
- *Plan metadata references*
- *Examples*
 - *Run another job*
 - *Slack channel message*
 - *Plan metadata examples*
 - *Feed metadata inputs to cloud function*
- *Verify Signatures*
 - *Signature Header*
 - *Check Application Tools*
 - *Process Signed Requests*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

During the execution of your plan, you can create a task to send HTTP requests to a third-party application endpoint. For example, when a preceding task successfully executes, you can send an HTTP message to a designated endpoint with information from that task.

- An **HTTP task** is a request between the Designer Cloud powered by Trifacta® platform and another application. These requests are delivered using over HTTP and can be interpreted by the receiving application to take action.

NOTE: Your receiving application may require that you whitelist the host and port number or IP address of the platform. Please refer to the documentation for your application.

- An HTTP task is one of the task types available in a plan. For more information, see *Plan View Page*.
- This capability may need to be enabled in your environment. For more information, *Overview of Orchestration*.

Limitations

- Custom security certificates cannot be used.
- HTTP-based requests have a 30-second timeout limit.

Prerequisites

NOTE: It's possible that webhook requests can be submitted back to the platform to execute API tasks within the platform. However, there are security concerns. Additional configuration is required. For more information, see *Configure Webhooks*.

Requirements for receiving application

To send an HTTP request to a target application, the application must be configured to receive the request:

- Requests from outside of the application domain must be enabled.

NOTE: Your receiving application may require that you whitelist the host and port number or IP address of the platform. Please refer to the documentation for your application.

- You must acquire the URL of the endpoint to which to send the HTTP request.
- You must acquire any HTTP headers that must be inserted with each HTTP request.
- If the request must be signed, additional configuration is required. Details are below.

Create Task


1. Open your plan in Plan View. In your sequence of tasks. Click a Plus sign icon to create a new task.

2. In the right panel, select **HTTP task**. The HTTP task panel is displayed.

<

HTTP task

×

 Send HTTP Request

Test

...

Method

GET

▼

URL

required

https://example.com/v4/connections

https://example.com/endpoint

Headers ?

Add

Authorization

<my_authorization_key>

Remove

Key

Value

Remove

Secret Key

Secret Key

☒ Validate SSL certificate

Retry

3

times

Cancel

Save

Figure: HTTP task

3. Set the required parameters. For more information on parameters, see *Plan View for HTTP Tasks*.
4. You can specify plan metadata information in the header values and request body of your request. For more information, see *Plan Metadata References*.
5. To test the connection, click **Test**. A success message is displayed.

Tip: A status code of 200 indicates that the test was successful.

Tip: You can use the GET method for testing purposes. A GET request does not change any data on the target platform but may permit you to specify elements in the request body.

6. To add the task, click **Save**.

Rename Task

To rename the task, click **More menu > Edit** in the right panel.

Tip: Good naming may include the target platform endpoint and method, as well as the purposes of the task in your plan.

Delete Task

To delete the task, click **More menu > Delete**. Confirm that you wish to delete the task.

This step cannot be undone.

Plan metadata references

Within the message of your other tasks, you can reference metadata about the plan, including this task. For more information, see *Plan Metadata References*.

Examples

Run another job

You can create a task to run another job on the successful execution of this one.

Tip: Use this method to create conditional sequences of job executions.

As needed, you can specify task overrides as part of launching a job via API. For more information, see *API Task - Run Job*.

Prerequisites

NOTE: For this example, the platform must be whitelisted to receive requests from itself. Additional configuration is required. For more information, see *Configure Webhooks*.

You must acquire the recipe identifier for the next job to execute.

1. Open the flow containing the next recipe.
2. In Flow View, click the recipe whose outputs you wish to generate.
3. Review the URL for the recipe object. In the example below, the recipe Id value is 4:

```
http://www.example.com:3005/flows/1?recipe=4&tab=recipe
```

4. Retain this value for below.

Define the HTTP task

Parameter	Description
-----------	-------------

Name	This name appears in the Designer Cloud application only.
Url	Specify the URL as follows, replacing the example values with your own: <div>http://www.example.com:3005/v4/jobGroups/</div>
Headers	Insert the following two headers: <div>key: Content-Type value: application/json</div> <div>key: Authorization value: Bearer <paste your access token here></div> <div>NOTE: The token value must be preceded by the string: Bearer.</div>
Body	In the body, insert the recipe Id for the value for wrangledDataset, which is the internal platform term for recipe: <div> <pre>{ "wrangledDataset": { "id": 4 } }</pre> </div>
Method	Select the POST method.

Verify

1. Run the plan for which the HTTP task was created.
2. When the plan successfully completes, open the flow containing the other job to execute.
3. When you select the target recipe, a new job should be queued, in-progress, or completed.

Slack channel message

Tip: Slack tasks are now a supported product feature. For more information, see *Create Slack Task*.

You can create an HTTP task to deliver a text message to a Slack channel of your choice.

Prerequisites

Set up your Slack installation to receive HTTP messages:

1. If needed, create a Slack channel to receive your messages.
2. Create an app.
3. Activate incoming HTTP messages for your app.
4. Specify the channel to receive your incoming messages.
5. Copy the URL for the incoming HTTP request from the cURL statement.

Define the HTTP task

Parameter	Description
-----------	-------------

Name	This name appears in the Designer Cloud application only.
Method	Select the POST method.
Url	Paste the URL that you copied from Slack.
Headers	Copy the content headers from the Slack cURL command: <pre>key: Content-Type value: application/json</pre>
Body	<pre>{"text": "Your job has completed."}</pre>

Verify

1. Click **Test** to validate that this task will work.
2. Run a job and check the Slack channel for a message.

Plan metadata examples

You can reference metadata information from the plan definition and the current plan run as part of the request of your HTTP task.

Notes:

- You can only insert metadata references for tasks that have already occurred in the plan run before the HTTP task begins.
- Each task in the current run is referenced using a two-letter code. Example:

```
{{ $http_xx.name }}
```

Syntax

A plan metadata reference is constructed using the following syntax. In the appropriate textbox, enter one of the following values:

Tip: Start by typing \$, which provides access to a menu tree of metadata references for each of the metadata reference types. The final syntax is noted above.

Plans:

Metadata information from the plan definition or the current plan run:

```
{{ $plan
```

Flows:

Metadata information for the flow tasks executed in the current plan run.

```
{{ $flow_
```

Flow task:

Metadata information for the outputs generated by the specific flow task.

```
{{flow_7p['My Output Name'].
```

In this example:

- `flow_7p` is a reference to the specific flow task.
- `'My Output Name'` is the display name for the underlying output.

Plan information

The following request body contains references to the Plan name, plan run identifier, and the flow that was just executed:

```
{ "text": "Plan: {{plan.name}}  
RunId: {{plan.runId}}  
Flow: {{flow_7p.name}}  
Success." }
```

Plan run information

The following request body contains plan execution information using timestamps:

```
{ "text": "Plan: {{plan.name}}  
RunId: {{plan.runId}}  
- plan start: {{plan.startTime}}  
Running time: {{plan.duration}}  
  
Times:  
- last task start: {{flow_7p.startTime}}  
- last task end: {{flow_7p.endTime}}  
" }
```

HTTP task information

You can reference information from an HTTP task that has already occurred:

```
{ "text": "{{http_qg.name}} returned {{http_qg.statusCode}}." }
```

Flow task information

The following request body references information from a flow task in the plan:

```
{ "text": "{{flow_7p.name}} execution:  
Duration: {{flow_7p.duration}}  
Status: {{flow_7p.status}}  
  
For more information, see jobIds: {{flow_7p.jobIds}}  
" }
```

Flow information

The following request body references information from the underlying output for the above flow task:

```
{
  "text": "Flow reference information:
  Name: {{$flow_7p['2013 POS'].name}}
  Favorite column: {{$flow_7p['2013 POS'].columns.Store_Nbr.name}}
  Least favorite data source: {{$flow_7p['2013 POS'].sources['POS-r01.txt'].name}}
  For more information, see jobIds: {{$flow_7p.jobIds}}
  "
}
```

Notes:

- You can reference columns from the generated results using the `.columns.` reference.
- You can reference information from datasources using the `.sources` reference.

For more information, see *Plan Metadata References*.

Feed metadata inputs to cloud function

This example demonstrates how you can use an HTTP task to deliver plan metadata to AWS lambda functions. A similar approach could be used for Google Cloud functions.

In this case, the `rowCount` value from the flow task execution is delivered via HTTP task to an AWS lambda function.

General steps:

1. Define your plan.
2. Flow task: Run the flow to generate the outputs needed for your Lambda function.
3. HTTP task: generates an HTTP request whose body includes a reference to the `rowCount` metadata variable. Request body:

```
{
  "rowCount": "{{$flow_7p['My Flow Name'].output['My output name'].rowCount}}"
}
```

4. **AWS Lambda functions:** The following is pseudo-code for Lambda:

```
import json
def lambda_handler(event, context):
    httpTaskBody = json.loads(event["body"])
    rowCount = httpTaskBody["rowCount"]

    return {
        'statusCode': 200,
        'body': json.dumps(rowCount)
    }
```

5. **Google Cloud functions:** The following is pseudo-code for Google Cloud functions:

```
def get_row_count(request):
    request_json = request.get_json()
    if request_json and 'rowCount' in request_json:
        rowCount = request_json['rowCount']
        return rowCount
    return 'No rowCount attribute provided'
```


Verify Signatures

Depending on the target application, implementing signature verification may require developer skills.

Optionally, you can configure the platform to sign the HTTP requests. Signed requests guarantee that the requests are sent from the platform, instead of a third party.

Below, you can review how the signature is created, so that you can configure the receiving application to properly process the signature and its related request.

Signature Header

HTTP requests are signed by inserting the `X-Webhook-Signature` header in the request. These signatures are in the following form:

```
X-Webhook-Signature: t=<timestamp>,sha256=<signature>
```

where:

- `<timestamp>` - Timestamp when the signature was sent. Value is in UNIX time.
- `<signature>` - SHA256 signature. The platform generates this signature using a hash-based message authentication code (HMAC) with SHA-256.

More information on these values is available below.

Example:

```
X-Webhook-Signature: t=1568818215724,sha256=55fa71b2e391cd3ccba8413fb51ad16984a38edb3cccfe81f381c4b8197ee07a
```

Check Application Tools

Depending on the application, you may need to complete one of the following sets of tasks to verify the task signatures:

NOTE: You may need to whitelist the platform in your application. See the application's documentation for details.

You may be required to create some custom coding for your application. Below, you can review details on how to do so, including a JavaScript example.

Process Signed Requests

Timestamp

The timestamp value (`t=<timestamp>`) appears at the beginning of the header value to prevent replay attacks, where an attacker could intercept a valid payload and its signature and re-transmit them.

- To avoid such attacks, a timestamp is included in the signature header and is also embedded as part of the signed payload.
- Since the timestamp is part of the signed payload, an attacker cannot change the timestamp value without invalidating the signature.
 - If the signature is valid but the timestamp is too old, you can then choose to reject the request.
 - For example, if you receive a request with a timestamp that corresponds to a date from one hour ago, you should probably reject the request.

- For more information on replay attacks, see https://en.wikipedia.org/wiki/Replay_attack.

Signature

The task signature includes as part of its hashed value:

- The secret key (entered above)
- The timestamp value
- Request data:
 - (POST/PUT/PATCH) - the body of the request
 - (GET/DELETE) - URL of the request

Step 1 - Extract the timestamp and signatures

Split the X-Webhook-Signature header:

1. Split values using the , character as a separator.
2. Split each of the parts using the = character.
3. Extract the values for the timestamp and signature. From the above example:
 - a. timestamp: 1568818215724
 - b. signature: 55fa71b2e391cd3ccba8413fb51ad16984a38edb3cccfe81f381c4b8197ee07a

Step 2 - Create the expected signature

In the receiving application, you can recompute the signature to verify that the request was sent from the platform.

1. Concatenate the timestamp, the dot character . and the request body (POST/PUT/PATCH methods) or the url (GET/DELETE methods).
2. Suppose the above example is the signature for a POST request, and the request body is test. The concatenated value is the following:

```
1568818215724.test
```

3. You can now compute the HMAC authentication code in your receiving application. In the following JavaScript example, the secret key value is mySecret:

```
const crypto = require('crypto');

const message = '1568818215724.test'; // as defined above

const hmac = crypto.createHmac('sha256', 'mySecret');
hmac.update(message)
const expectedSignature = hmac.digest('hex');
```

Step 3 - Compare the signatures

The value returned by your code and the value included as the signature in the X-Webhook-Signature header should be compared:

- If the values do not match, reject the request.
- If the values do match, compute the difference between the current timestamp and the timestamp in the header. If the difference is outside of your permitted limit, reject the request.
- Otherwise, process the request normally in your application.

Create Slack Task

Contents:

- *Limitations*
 - *Prerequisites*
 - *Create Task*
 - *Rename Task*
 - *Delete Task*
 - *Plan metadata references*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create plan tasks to deliver messages to accessible Slack channels. These tasks are helpful for informing a set of stakeholders across your organization about the execution of your plans.

- A **Slack task** is a message from the Designer Cloud powered by Trifacta® platform to a specified Slack workspace channel.
- A Slack task is one of the task types available in a plan. For more information, see *Plan View Page*.
- This capability may need to be enabled in your environment. For more information, see *Overview of Orchestration*.

Limitations

- You can only post messages to Slack channels. Other interaction methods are not supported.

Tip: You can also create HTTP tasks to deliver messages to a Slack channel. See *Create HTTP Task*.

- HTTP-based requests have a 30-second timeout limit.
- Authentication must be made through OAuth.

Prerequisites

- To send a message to Slack, you must create an app in the target workspace for the Slack channel to receive the message. For more information, see <https://api.slack.com/apps>.
- This Slack app must support OAuth authentication. The OAuth Token that you create must be installed in your workspace.

NOTE: Copy the generated token to a text file and retain it for later. This token must be pasted into the definition of each Slack task where you wish to use it.

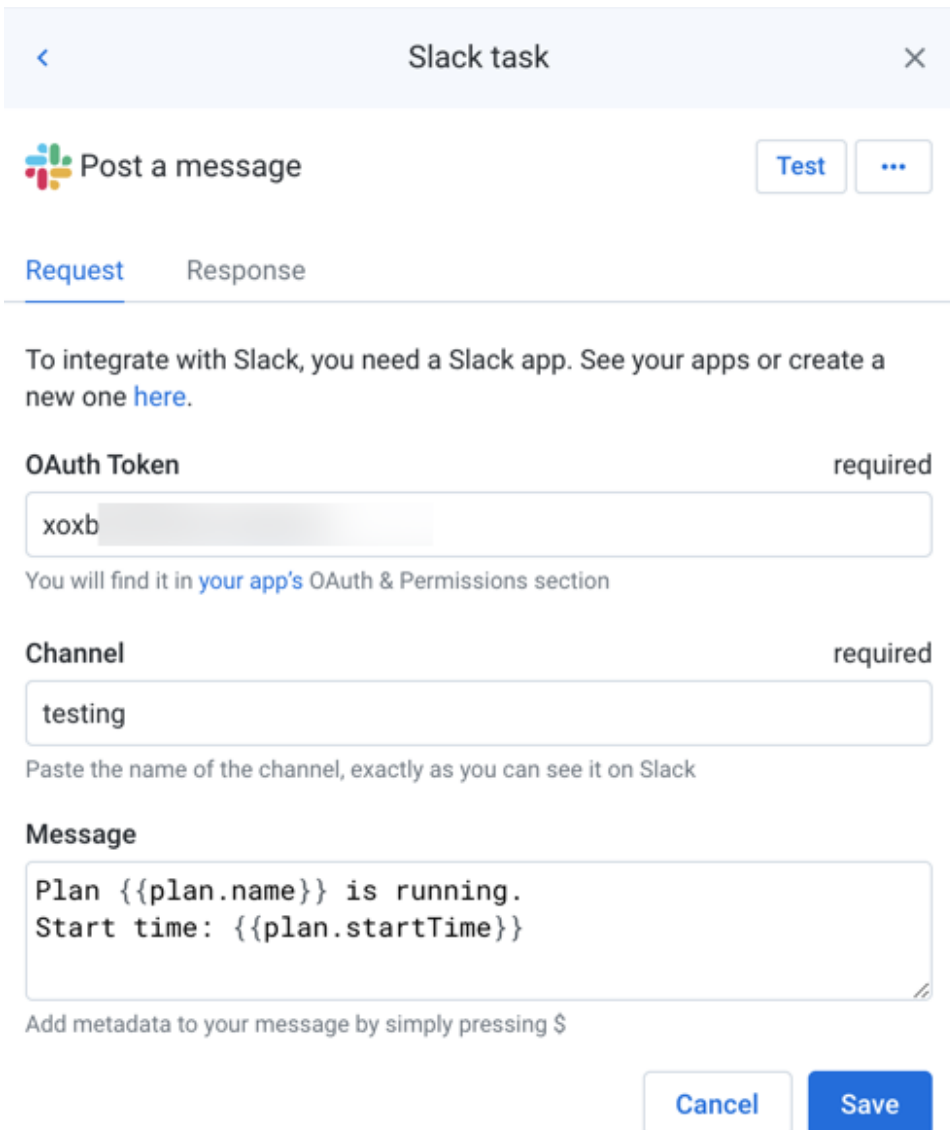
- Create an OAuth Token that has `chat:write` scopes. This token is inserted into your task definition. There are two types of tokens:
 - **Bot Token:** These tokens post a Slack message from the name of the app.
 - The Bot Token also requires the `chat:write.public` scope.
 - A Bot Token is required if you wish to send a direct message through the App category of messages.

- To send a message to a private channel using a Bot Token, you must install the app in the channel through the Integrations window in the channel's settings.
- **User Token:** These tokens post a Slack message from the user who authorizes the message.

Tip: To send a message to a user or a private channel using a User Token, additional configuration is required. See below.

Create Task

1. Open your plan in Plan View. In your sequence of tasks. Click a Plus sign icon to create a new task.
2. In the right panel, select **Slack task**. The Slack task panel is displayed.



The screenshot shows a configuration window titled "Slack task" with a close button (X) in the top right corner. Below the title bar, there is a "Post a message" section with a Slack logo icon, a "Test" button, and a three-dot menu button. Below this is a tabbed interface with "Request" (selected) and "Response" tabs. The main content area contains instructions: "To integrate with Slack, you need a Slack app. See your apps or create a new one [here](#)." It then lists two required fields: "OAuth Token" (with a text input containing "xoxb" and a note "You will find it in [your app's](#) OAuth & Permissions section") and "Channel" (with a text input containing "testing" and a note "Paste the name of the channel, exactly as you can see it on Slack"). Below these is a "Message" section with a text area containing the template "Plan {{plan.name}} is running. Start time: {{plan.startTime}}". A note below the text area says "Add metadata to your message by simply pressing \$". At the bottom right are "Cancel" and "Save" buttons.

Figure: Slack task

3. In the Request tab, set the required parameters. For more information, see *Plan View for Slack Tasks*.
4. To test the message, click **Test**. A success message is displayed.

Tip: A status code of 200 indicates that the test was successful.

5. To add the task, click **Save**.

Rename Task

To rename the task, click **More menu > Edit** in the right panel.

Tip: Good naming may include the target platform endpoint and method, as well as the purposes of the task in your plan.

Delete Task

To delete the task, click **More menu > Delete**. Confirm that you wish to delete the task.

This step cannot be undone.


Plan metadata references

Within the message of your Slack task, you can reference metadata about the plan that is being executed. For more information, see *Plan Metadata References*.

Diagnose Failed Plan Runs

Contents:

- *Identify Plan Run Failures*
 - *Definition*
 - *Failure messages*
- *Retry Failed Plan Runs*
- *Logs*
- *By Task Type*
- *Contact Support*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

This section provides information and references on how to identify and repair issues that have led to failures in a plan run.

Identify Plan Run Failures

Definition

A plan execution has failed if any individual task in the plan run fails. If so, the plan run is marked as failed in the application.

After a task has failed, plan execution could still continue if the plan logic includes `On failure` tasks from the task that did fail.

Failure messages

When a plan run fails to complete, a failure message appears in following locations:

- **Plans Page:** Listing for the plan.
- **Plan Runs Page:** Listing of individual plan runs.

The following is an example from the Plan Runs page:








Plan runs				
All runs Completed Failed Canceled Running				 <input type="text" value="Search plan runs"/>
		Status	User	Started ▾
>	 Orchestration_Sanity Run ID: 11	 Failed	Test User 6492291878	Today at 2:39 AM  Ran for a few seconds
>	 Orchestration_Sanity Run ID: 10	 Completed	Test User 6492291878	Today at 1:39 AM  Ran for a few seconds

Figure: Plan run failed

Retry Failed Plan Runs

When a plan run fails, you can rerun the plan You can retry failed plan runs by clicking the **Retry from failed** button. For more information, see *Plan Run Details Page*.

Logs

Logs for plan runs are available in `orchestration-service.log` file in the support bundle. From the Resources menu, select **Download logs**. For more information, see *Support Bundle Contents*.

By Task Type

NOTE: A plan run is essentially a sequence of tasks that

are executed based on the outcome(s) of previous tasks. Failures are likely to be the result of failures to execute individual tasks. Please use the links below to explore information for specific tasks.

The following types of tasks can be executed as part of a plan in the Designer Cloud powered by Trifacta® platform :

- **Flow task:** An ad-hoc or scheduled execution of the transformations required to produce one or more selected outputs from a flow.
 - **Snapshot:** An image of the flows, recipes, and datasets of the flow tasks in a plan, taken at the time of plan execution.
 - **Trigger:** A trigger is a scheduling object that launches a plan run at a user-defined time.
- **HTTP task:** A request submitted to a third-party server as part of a plan run.
- **Slack task:** A specialized version of HTTP task, a Slack task can be configured to delete a message to a specified Slack channel.
- **Delete task:** You can specify a task to delete one or more files on backend storage. This task type is useful for removing intermediate files that are generated as part of job or plan execution.

For more information, see *Plan View Page*.

For more information on plans in general, see *Overview of Orchestration*.

Contact Support

If you are unable to diagnose your plan run failure, please contact *Alteryx Support*.

NOTE: When you contact support about a plan run failure, please be sure to download and include the support bundle. See *Download Logs Dialog*.

Export Plan

This section provides an overview on how to export plans from one environment to another environment as a ZIP file.

You can export a plan from the Designer Cloud powered by Trifacta® platform from one system to another system as a zip file. The exported zip file contains a JSON file for the plan and JSON files for each referenced asset. The plan and its tasks are exported into the plan definition file.

NOTE: When you export a plan that has flow tasks, then all the corresponding flows are also exported.

For each flow included in the plan:

- Each flow is exported only once in a flow definition file, even if it is used in many plan flow tasks.
- If the flow contains any artifact files, they are included as `.data` files next to the plan definition file. These files should be imported with the flow, too.

NOTE:

- You can unzip the exported plan ZIP, remove any of the flow files, and re-zip if you want to import the plan into the same workspace without replicating the flows.
- You can upload the plan as a single JSON file without re-zipping, if there are no flow files for the plan.

When you export the plan ZIP file, a snapshot of the plan is taken at the time of export.

Plan exports are useful for:

- Backing up the work in progress on your plans
- Archiving of completed development work

Limitations

- The plan file does not include the plan schedules and their associated overrides.
- Import and export of plans is not supported in Deployment Manager.

Export from Plans page

Steps :

1. From the home page of the Designer Cloud powered by Trifacta platform , navigate to **Plans**.
2. In the Plans page, select the required plan. From the context menu, select **Export**.
3. In the Export Plan window, select **Download package(.zip)**.
4. Enter any optional notes, if required.
5. The ZIP file is downloaded to the default download location of your desktop.

Tip: You can also export a plan from Plan View page.

Import Plan

An exported plan can be imported into the Designer Cloud powered by Trifacta® platform into a different workspace.

Limitations

- You cannot import a plan that was exported in an earlier release.
- You cannot import schedules while importing a plan.
- You cannot modify plan definitions outside of the Designer Cloud application .

NOTE: After importing a plan, the objects referenced in the plan must be connected to the corresponding resources available in the target system.

NOTE: When you import a plan, the corresponding flow tasks and HTTP tasks in the plan are also imported.

NOTE:

- When you import a plan, you must import a ZIP file containing the JSON definition and any included flows.
- You can import the plan as a single JSON file without copying the flow files, as it reuses the existing flow files in the workspace.

Import

Steps:

1. From the home page of the Designer Cloud powered by Trifacta platform , navigate to **Plans**.
2. From the context menu of the Plans page, select **Import**.

Tip: You can import multiple plans (ZIP files) through the file browser or through drag-and-drop. Press **CTRL/CMD** + click or **SHIFT** + click to select multiple files for import.

3. From the Import Plans window, select the exported zip file from your system and click **Open**.

The plan is imported and available for use in the Plans page.

Share a Plan

Contents:

- *Limitations*
 - *Permissions*
 - *Steps*
-

This section provides an overview of sharing plans with other users for collaboration on the same plan.

You can share plans with one or more users to work together on the same plan. You can share the plan through the Plans page.

Limitations

- When a plan is shared, the assets of the underlying tasks are not shared directly.
- The plan can be executed only if the user has access to all the underlying tasks.
- Plan schedules cannot be shared with users.

Permissions

When a user is provided access to a plan, the user becomes a **collaborator** on the plan and is assigned a subset of the permissions assigned to the **owner** of the plan. If the user has minimal permissions for overall plans then sharing the plan as collaborator would be downgraded.

NOTE: A collaborator on a plan cannot delete the plan.

NOTE: In addition to the shared plan, you must have collaborator access to all underlying assets to execute a plan.

For more information, see *Overview of Sharing*.

Steps

1. In the left nav bar, click the **Plans** icon.
2. In the Plans page, hover over the plan to share.
3. Click the **Share** button that appears.

NOTE: You must be the plan owner or an admin to share a plan. Plan collaborator may share depending on the workspace-level access assigned to their role.

NOTE: Users who have been shared plans must have share-level access to the underlying assets in order to execute the tasks in the plan. Plan schedules cannot be shared.

4. In the Share dialog:

- a. Enter the email addresses of one or more users with whom you wish to share. Users must be members of the project or workspace.

Tip: You can paste in a comma-separated list of email addresses.

- b. From the drop-down, select the level to which the user or users has access.
5. To share the plan with the listed users at the selected access level, click **Share**.

Plan Metadata References



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Contents:

- *General Syntax*
- *\$plan References*
- *\$http References*
 - *Response references*
- *\$slack References*
- *\$delete References*
- *\$flow References*
 - *Output references*
- *Exploring Metadata*
 - *Metadata functions*
 - *Metadata structure*
- *Parameter References*
 - *Flow parameters*
 - *Environment parameters*
- *Additional References*

In the body and header of HTTP tasks in your plans, you can reference the following elements of metadata from the plan run for additional contextual information.

General Syntax

All plan metadata references follow the following basic syntax:

```
{{ $plan.path.to.reference }}
```

- All references can be entered with \$ in the Designer Cloud application . These references are turned into { \$ in the code definition. The double-curly braces forms the environment for metadata replacement.

Tip: In the Designer Cloud application , you can start by typing \$.

- Nodes in the tree are separated with a . period.

Reference values that contain whitespace must be listed in the following manner:

```
{{ $plan.path['path with white space in it'].rest.of.path }}
```

Notes:

- In the Designer Cloud application , you can use double-quotes when specifying a whitespace value. However, these double-quotes get escaped in the actual request. It is safer and more consistent to use single quotes.

Whitespace values typically appear when referencing the display name values for underlying assets.

\$plan References

These references apply to the plan definition or current plan run.

Text to enter:

```
$plan.
```

Reference	Description
name	Name of the plan that is run.
duration	Length of time that the plan ran or has run so far <div>Tip: To return a more readable form of this duration value, use the following reference: <pre>{{ \$plan.duration humanizeDuration }}</pre></div>
startTime	Timestamp for when the plan run began
runId	Internal identifier for this run of the plan
user	Internal identifier of the user who launched this run.
taskCount	Count of tasks in the plan run.

\$http References

These references apply to HTTP tasks in the plan run.

Enter the following, after which you can see the two-letter codes for the HTTP tasks that have already executed in the current plan run:

```
$http_ax.
```

Reference	Description
name	Name of the HTTP task
status	Current status of the task execution
duration	Length of time that the task ran or has run so far
startTime	Timestamp for when the task began. A null value if the task has not begun.
endTime	Timestamp for when the task ended. A null value if it has not ended yet.
statusCode	Status code (if any) returned from the receiving endpoint
response	Response information. See below.

Response references

These references apply to the response returned as part of the task execution.

Enter the following, after which you can see the two-letter codes for the HTTP tasks that have already executed in the current plan run:

```
$http_ax.response.
```

Reference	Description
body	Body of the response
json	JSON-formatted version of the response
headers	Headers returned with the response

\$slack References

You can reference metadata from Slack tasks in the current plan run using the following reference types:

```
$slack_ax.
```

Supported metadata is identical to the metadata for HTTP tasks. See the previous section for details.

\$delete References

You can reference metadata from Delete tasks in the current plan run using the following reference types:

```
$delete_92.
```

Supported metadata is identical to the metadata for HTTP tasks. See the previous section for details.

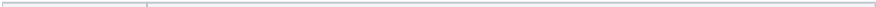
Reference	Description
name	Name of the HTTP task
status	Current status of the task execution
path	Full path to the file(s) or folder that was deleted.
duration	Length of time that the task ran or has run so far
startTime	Timestamp for when the task began. A null value if the task has not begun.
endTime	Timestamp for when the task ended. A null value if it has not ended yet.
deletedFilesCount	Number of files that were successfully deleted.

\$flow References

These references apply to flow tasks in the plan run.

Enter the following, after which you can see the two-letter codes for the HTTP tasks that have already executed in the current plan run:

```
$flow_ax.
```



Reference	Description
name	Name of the flow task
status	Current status of the task execution
duration	Length of time that the task ran or has run so far
startTime	Timestamp for when the task began. A null value if the task has not begun.
endTime	Timestamp for when the task ended. A null value if it has not ended yet.
jobIds	Internal identifiers for the jobs that were run as part of this flow task
flowName	Name of the flow underlying this flow task
output	Metadata from the flow task's output. See below.
params	Parameters created in the flow can be referenced in the task.

Output references

These references apply to the outputs that are generated in the flow tasks of the plan run.

Enter the following for flow task 7p with output My Output Name:

```
$flow_7p['My Output Name'].
```

Reference	Description
name	Name of the flow
status	Current status of the flow
duration	Length of time that the flow execution ran or has run so far
startTime	Timestamp for when the flow execution began. A null value if the run has not begun.
endTime	Timestamp for when the flow execution ended. A null value if it has not ended yet.
lastUpdate	Timestamp for when the flow was last modified
jobIds	Internal identifier(s) for the job that was run or is running for the flow. Can contain multiple identifiers.
user	Internal identifier for the user who executed the job
jobType	The type of job that was executed. Values: <ul style="list-style-type: none"> • <code>manual</code> - executed through the Designer Cloud application • <code>scheduled</code> - executed according to a defined schedule • <code>api</code> - executed via API
fileSize	If the output generates a file or files, this value captures the size in KB of the output.
environment	Running environment where the job was executed
columnCount	Count of columns generated in the output
rowCount	Count of rows generated in the output
dataTypesCount	Count of Trifacta data types detected in the output
validValuesCount	Count of valid values in the output
mismatchedValuesCount	Count of mismatched values in the output
emptyValuesCount	Count of missing or empty values in the output

columns	Column information from the selected output for the flow. See below.
sources	Source filename and table information from the imported datasets. See below.
publishing actions	Data on the publishing actions defined for the output. See below.

Output column references

Tip: The metrics calculated for output columns may also appear in profiles of your output data.

These references are available for output columns in the following syntax:

```
$flow_7p['My Output Name'].output.
```

Reference	Description
name	Column name
type	Data type of column
validValuesCount	Count of valid values in the column
mismatchedValuesCount	Count of mismatched values in the column
emptyValuesCount	Count of empty values in the column
topValues	List of top values in the column
minimumValue	Lowest value in the column
lowerQuartileValue	25th percentile value in the column
medianValue	50th percentile value in the column
upperQuartileValue	75th percentile value in the column
maximumValue	Maximum value in the column

Sources references

These references apply to the datasource files or tables that were used to generate the output.

Reference	Description
name	Name of the datasource file or table

Publishing actions references

These references apply to the publishing actions defined for the output.

Reference	Description
name	Name of the publishing action
action	Definition of the publishing action
location	Location where the publishing action is published
type	Type of publishing action

Exploring Metadata

You can use the following functions and techniques to further explore the metadata returned from your plan execution.

Metadata functions

The following functions can be applied to select metadata references to further filter the raw values.

humanizeDuration

In raw form, the `duration` metadata references return values that look like the following:

```
PT7.523S
```

You can apply the `HumanizeDuration` function to render the above into a more readable format:

```
{{ $plan.duration | humanizeDuration }}
```

The output of the above applied to the first value is the following:

```
7.523 seconds
```

uuid

You can generate a universally unique identifier, which can be delivered as part of a messaging payload:

```
{{ uuid() }}
```

Metadata structure

Some metadata references return complex or nested objects, which may return data that looks like the following:

```
[object Object],[object Object],[object Object],[object Object],[object Object],[object Object],[object Object]
```

In these cases, the nested data inside the object is not exposed by the basic reference. To explore further, you can use either of the following solutions:

Text: Create a for loop through the returned objects:

```
{% for value in $flow_6f.Job.columns.MyColumnName.topValues %}{{value.key}}, {{value.count}} {% endfor %}
```

JSON: add the `| dump` command to the end of your reference. You can modify the following example reference and try to insert in the Body textbox:

```
{{ $flow_6f.Job.columns.MyColumnName.topValues | dump | replace('"', '\\"') }}
```

Tip: When the data is returned, you can use loop structures to retrieve specific values for display.

For more information, see <https://mozilla.github.io/nunjucks/templating.html#dump>.

Parameter References

Flow parameters

To reference flow parameters as part of a flow task, use the following pattern:

```
$plan.params.<myParamName>
```

Environment parameters

You can reference environment parameters by name in your plan task metadata. For example:

```
env.MyEnvironmentParameter1
```

Additional References

Plan metadata reference information leverages the Nunjucks templating language, which provides additional capabilities such as loops, conditions, filters, and helper functions.

NOTE: These additional capabilities are available through the language, but their implementation in the Designer Cloud application has not been certified. For Nunjucks capabilities not listed on this page, you should experiment with them in a development environment first.

For more information, see <https://mozilla.github.io/nunjucks/templating.html>.

Library

Contents:

- *Basics*
 - *Overview*
 - *Interface*
 - *How To*
-

In the Library, you can create or import assets for use in the Designer Cloud powered by Trifacta® platform .

Basics

For more information, see *Import Basics*.

Overview

You can import data for use in the Designer Cloud powered by Trifacta platform through a reference object called an **imported dataset**. An imported dataset is a reference to the source of the data.

NOTE: The source data is never modified. In some cases, the source data may be copied to the base storage layer. For example, data that is uploaded from your local desktop must be copied to the base storage layer so that it is accessible to you and potentially other users of the Designer Cloud powered by Trifacta platform .

For more information, see *Overview of Storage*.

Interface

For more information, see *Library Page*.

How To

For getting started with tasks, see *Library Tasks*.

Library Page

Review the assets to which you have access in the Library page.

Tip: If you land in an empty Library page, you can start adding datasets. Click **Import Data**. See *Import Data Page*.

Library

All Data

Imported Datasets

References

Macros

All Data

Import Data

Find Datasets... (/)

AllOwned by meShared with me








Name	In Flows	Source	Last Updated ▾
 Dataset with Parameters	0	HDFS	Today at 11:33 AM
 POS-schema.csv	1	HDFS	Today at 11:28 AM
 POS-r03.txt	1	HDFS	Today at 11:28 AM
 POS-r01.txt	1	HDFS	Today at 11:28 AM
 REF_CAL.txt	1	HDFS	Today at 11:28 AM
 POS-r02.txt	1	HDFS	Today at 11:28 AM
 REF_PROD.txt	1	HDFS	Today at 11:28 AM

Figure: Library Page

Tabs:

- **All Data:** You can view all the imported datasets or references available to you.
- **Imported Datasets:** Review your imported datasets from sources such as file-based storage, connected databases, or desktop.
 - The Source column indicates where the original source data is located.
- **References:** Reference datasets are created from a recipe's output.
- **Macros:** Macros are sequence of steps that can be reused in other's recipe.

Filter by type:

Click one of the pre-defined filters to show datasets of the following types:

Filter by ownership:

For the selected object type, you can filter based on the ownership of the object:

- **All:** All objects of the selected type to which you have access.
- **Owned by me:** All objects of the selected type that you own.
- **Shared with me:** All objects of the type that have been shared with you.

Columns:

- **Name:** Name of the object.
- **Owner:** Owner of the object.
- **In flows:** Count of flows in which the object is in use.
- **Source:** Flow or datastore where the object is located.
- **Last updated:** Timestamp of the last time that the object was modified.

Actions:

- **Browse:** If displayed, use the page browsing controls to explore the available objects.
- **Search:** To search object names, enter a string in the search bar. Results are highlighted immediately in the Library page.
- **Sort:** Click a column header to sort the display by the column's entries.

Object Actions:

Hover over an object to reveal these actions on the right side of the screen.

- **Details:** Review details about the dataset. See *Dataset Details Page*.
- **Preview:** Inspect a preview of the dataset.

NOTE: Preview is not available for binary format sources.

- **Use in new flow:** (Imported dataset only) You can create a new flow and begin immediately wrangling the dataset. This step also creates a recipe in the flow.
- **Add to flow:** Add the dataset to a new or existing flow.
- **Make a copy:** Create a copy of the imported dataset. This option is not available for reference datasets.
- **Edit name and description:** Change the name and description of the dataset.
- **Edit data settings:** If the source of the imported dataset required conversion to an internally supported format, you can modify settings related to that conversion process. For more information, see *File Import Settings*.

Tip: This setting applies primarily to binary file formats, such as PDF and Excel, or file formats that may require additional steps to convert into tabular data, such as JSON.

- **Refresh dataset:** If available, this option refreshes the dataset's metadata with the latest source schema.

NOTE: When a dataset is refreshed, all samples associated with the dataset are deleted, whether the dataset has changed. Samples must be recreated in their recipes.

NOTE: If you attempt to refresh the schema of a parameterized dataset based on a set of files, only the schema for the first file is checked for changes. If changes are detected, the other files are contain those changes as well. This can lead to changes being assumed or undetected in later files and potential data corruption in the flow.

For more information, see *Overview of Schema Management*.

- **Transfer ownership:** For assets that you own, you can transfer ownership of them to another user. For more information, see *Transfer Asset Ownership*.
- **Delete dataset:** Delete the dataset.

Deleting a dataset cannot be undone.

Imported Datasets

NOTE: You can only see the imported datasets to which you have access in your currently selected project or workspace. If the data underlying the imported dataset is not available, the imported dataset is still listed in the Library page, since it is just a reference to the data.

To create a new imported dataset, click **Import Data**. For more information, see *Import Data Page*.

For more information, see *Imported Datasets Page*.

References

A reference dataset is a reference to a recipe's output. For more information, see *References Page*.

NOTE: A reference dataset is a read-only object where it is referenced. A reference dataset must be created in the source flow from the recipe to use.

A reference dataset is created from the context menu of a flow's recipe.

Macros

A macro is a saved sequence of one or more recipe steps that can be reused in other recipes. See *Macros Page*.

You can either import macros from your desktop or browse through the Designer Cloud powered by Trifacta® platform community page for existing macros. For more information, see *Import Macro*.

Imported Datasets Page

You can review the imported datasets that you own or have been shared with you.

NOTE: If the data underlying the imported dataset is not available, the imported dataset is still listed in the Library page, since it is just a reference to the data.

Tip: To create an imported dataset, click **Import Data**. For more information, see *Import Data Page*.

Imported Datasets				
Import Data				
1 2 3 4 5				
Search Datasets				
All Owned by me Shared with me				
Name	Owner	In flows	Source	Last updated
Advertising_Clickstream_wal...		1	TFS	Yesterday at 1:17 PM
Advertising_Clickstream_wal...		1	TFS	05/24/2022
Dataset-ProductNames-Order...		2	TFS	04/19/2022
Dataset-ProductNames.csv		1	TFS	04/19/2022
Reference-ProductNames		1	TFS	04/19/2022
april_challenge_source.csv		1	TFS	03/16/2022
Copy of WI_Home_Grown_Far...		0	TFS	03/04/2022
Groups		1	Zendesk-doc	03/04/2022

Figure: Imported Datasets Page

You can filter the imported datasets based on the ownership of the datasets.

- **All:** All datasets which you have access.
- **Owned by me:** All datasets that you own.
- **Shared with me:** All datasets that have been shared with you.

Columns:

- **Name:** Name of the imported dataset.
- **Owner:** Owner of the imported dataset.
- **In flows:** Count of flows in which the imported dataset is in use.
- **Source:** Source of the data for the imported dataset.
- **Last Updated:** Timestamp of the last time that the imported dataset was modified.

For large relational or Parquet datasets, you can monitor the import process through the Library page.

- During the import process, you can hover over the icon for a **pending dataset** to track status.
- Click the icon for additional details. See *Dataset Details Page*.

NOTE: This feature may require enablement in your deployment. See *Configure JDBC Ingestion*.

For more information, see *Overview of Job Monitoring*.

Actions:

- **Browse:** If displayed, use the page browsing controls to explore the available objects.
- **Search:** To search object names, enter a string in the search bar. Results are highlighted immediately in the Library page.
- **Sort:** Click a column header to sort the display by the column's entries.

Context menu options:

NOTE: The context menu options vary based on the ownership of the datasets.

- **Preview:** Inspect a preview of the dataset.

NOTE: Preview is not available for binary format sources.

- **Use in new Flow:** (Imported dataset only) You can create a new flow and begin immediately wrangling the dataset. This step also creates a recipe in the flow.
- **Add to Flow:** Add the dataset to a new or existing flow.
- **Make a copy:** Create a copy of the imported dataset. This option is not available for reference datasets.
- **Edit name and description:** Change the name and description of the dataset.
- **Remove Structure:** Removes the structure of the dataset and converts into a raw dataset.
- **Refresh Dataset:** If available, this option refreshes the dataset's metadata with the latest source schema.

NOTE: If you attempt to refresh the schema of a parameterized dataset based on a set of files, only the schema for the first file is checked for changes. If changes are detected, the other files are contain those changes as well. This can lead to changes being assumed or undetected in later files and potential data corruption.

- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user. See *Transfer Asset Ownership*.
- **Delete Dataset:** Delete the dataset.

Deleting a dataset cannot be undone.

Dataset Details Page

Contents:

- Imported Dataset
- Reference Dataset
- Dataset with Parameters
- Dataset with SQL

Use the Dataset Details page to review a dataset's usage and to perform management tasks on it.

Imported Dataset

For datasets that have been imported into the Designer Cloud powered by Trifacta® platform , you can review source location and current usage within flows to which you have access.

Status: For large relational datasets, you can track status of the import process. For more information, see *Overview of Job Monitoring*.

NOTE: This feature may require enablement in your deployment. See *Configure JDBC Ingestion*.

allTypes_2.pqt

Last updated: Today at 5:37 PM
File size: 28.81kB
Location:

Created: Today at 5:37 PM
Size: 17 columns - 7 types
Column data type inference: Enabled

Use in new Flow

Preview

...


Used in 1 Flow

Name	Owner	Objects	Last updated
<div><div>SQLUSER_TEST20220119173730</div><div>API TestFramework - Flow.</div></div>	Administrator	1 Dataset, 1 Recipe	Today at 5:37 PM

Figure: Imported dataset details

Actions:

- **Use in new flow:** Create a new flow for your dataset and begin wrangling.
- **Preview:** Review the first few rows of the dataset.
- **Add to flow:** Add imported dataset to a new or existing flow.
- **Make a copy:** Create a copy of the imported dataset.
- **Edit name and description:** Edit the name and description for the dataset.
- **Remove structure:** Remove initial steps applied to structure data.
- **Refresh dataset:** If available, this option refreshes the dataset's metadata with the latest source schema.

 Dataset with Parameters

Wrangle in new Flow
 Preview
 ...

Last updated: Today at 11:33 AM
 Created: Today at 11:33 AM
 Size: 16 columns · 4 types

Parameters

Path: `eed8f72c-edcc-40f6-b067-797841e1cc1c/POS-r_*_{digit}{digit}.txt`

Parameters	Type	Default value	Name
<code>*_{digit}{digit}</code>	Pattern	matches against trifacta pattern: <code>'{digit}{digit}'</code>	

Figure: Dataset with Parameters details

You can review the parameters and variables that have been defined for the dataset.

Action:

- **Use in new flow:** Create a new flow for your dataset and begin wrangling.
- **Preview:** Review the first few rows of the dataset.
- **Add to flow:** Add imported dataset to a new or existing flow.
- **Make a copy:** Create a copy of the imported dataset.
- **Edit name and description:** Edit the name and description for the dataset.
- **Edit parameters:** Modify the parameters used to create the dataset. See *Create Dataset with Parameters*.
- **Remove structure:** Remove the initial parsing structure. When the structure is removed:
 - The dataset is converted to an unstructured dataset. An **unstructured dataset** is the source data converted into a flat file format.
 - All steps to shape the dataset are removed. You must break up columns in manual steps in any recipe created from the object.
- **Refresh dataset:** If schema refresh is enabled, you can use this option to check for updates to the schema of the dataset source.

Actions for dataset owners:

- **Transfer ownership:** Transfer ownership of this asset to another users. See *Transfer Asset Ownership*.
- **Delete dataset:** Delete the dataset.

Deleting a dataset cannot be undone.

Dataset with SQL

For datasets that were created with SQL statements, the following details are available.

Through the custom SQL interface, it is possible to enter SQL statements that can delete data, change table schemas, or otherwise corrupt the targeted database. Please use this feature with caution.

NOTE: If you modify the SQL statement for your imported dataset, any samples based on the old SQL statement are invalidated.

SQL Dataset 2

Use in new flow

Preview

Edit custom SQL

...

Last updated: 10/21/2022
Connection: Snowflake
Status: Completed • 26 sec
Custom SQL [Edit](#)

Created: 10/21/2022
Column data type inference: Enabled
Job ended: 10/21/2022

Parameters

Parameters	Name	Default value	Type
table_name_tail	table_name_tail	Matches "22767474940950"	Variable

Used in 1 Flow

Name	Owner	Objects	Last updated
SPO-DatasetWithParamsTest	Steve Olson	1 Dataset, 1 Recipe	Today at 5:50 AM

Figure: Dataset With SQL Details

Actions:

- **Use in new flow:** Create a new flow for your dataset and begin wrangling.
- **Preview:** Review the first few rows of the dataset.
- **Edit custom SQL:** Edit the custom SQL that has been used to define the dataset. For more information, see *Create Dataset with SQL*.
- **Add to flow:** Add imported dataset to a new or existing flow.
- **Make a copy:** Create a copy of the imported dataset.
- **Refresh dataset:** Refresh the dataset.

Actions for dataset owners:

- **Transfer ownership:** Transfer ownership of this asset to another users. See *Transfer Asset Ownership*.
- **Delete dataset:** Delete the dataset.

Deleting a dataset cannot be undone.

Import Data Page

Contents:

- General Limitations
- Basic Task
 - 1. Connect to sources
 - 2. Add datasets
 - 3. Configure selections
 - 4. Import selections

Through the Import Data page, you can upload datasets or select datasets from sources that are stored on connected datastores. From the Library page, click **Import Data**.

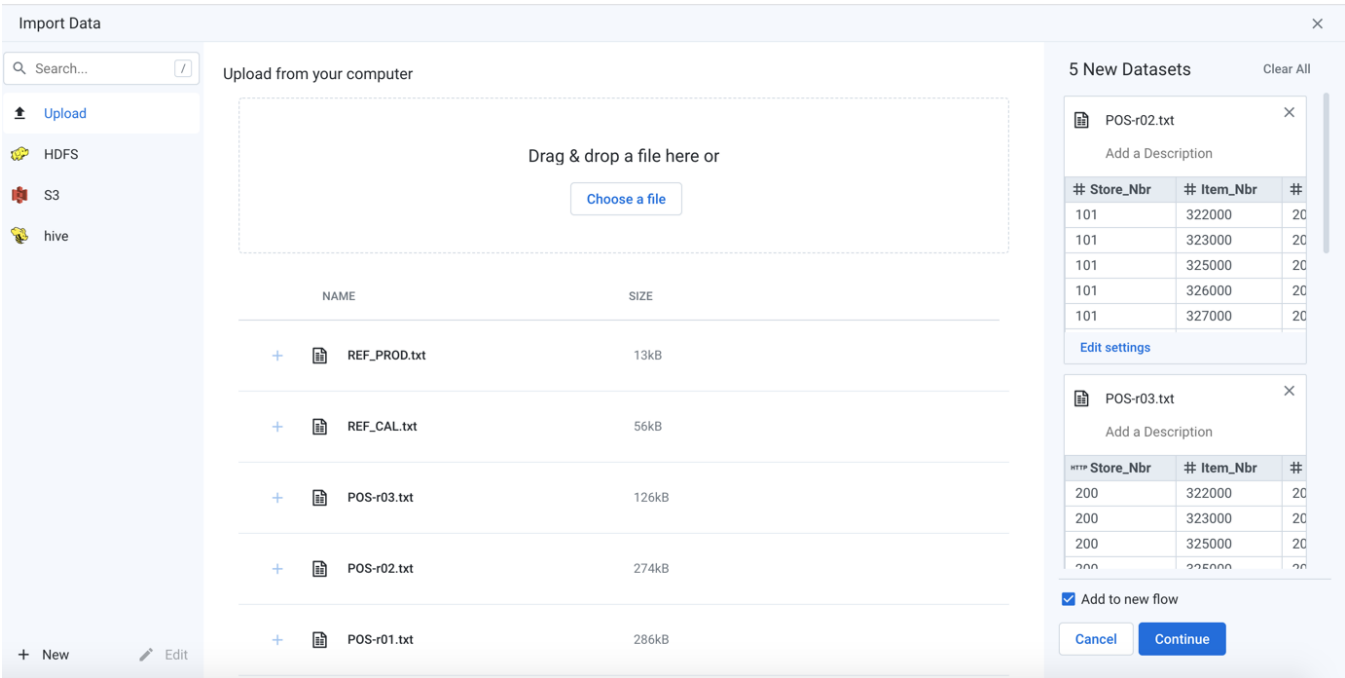


Figure: Import Data page

General Limitations

NOTE: For file-based sources, the Designer Cloud powered by Trifacta® platform expects that each row of data in the import file is terminated with a consistent newline character, including the last one in the file.

- For single files lacking this final newline character, the final record may be dropped.
- For multi-file imports lacking a newline in the final record of a file, this final record may be merged with the first one in the next file and then dropped in the Trifacta Photon running environment.



NOTE: To be able to import datasets from the base storage layer, your user account must include the `dataAdmin` role.

NOTE: An imported dataset requires about 15 rows to properly infer column data types and the row, if any, to use for column headers.

File and path limitations:

- The colon character (`:`) cannot appear in a filename or a file path.
- Filenames cannot begin with special characters like dot (`.`) or underscore (`_`).
- Input file or table paths can have a maximum length of 1024 characters.

Basic Task

1. Connect to sources

During import, the Designer Cloud application identifies file formats based on the extension of the filename.

- Compressed files are recognized and can be imported based on their file extensions.
- Filenames that do not have an extension are treated as TXT files.

Upload: The Designer Cloud powered by Trifacta platform can also load files from your local file system.

Tip: You can drag and drop files from your desktop to to upload them.

NOTE: You can upload a file up to 1 GB in size.

NOTE: When you upload an updated version of a previously uploaded file, the new file is stored as a separate upload altogether. Where the imported dataset based on the previous version is used, you must swap out the old dataset to point to the new one.

HDFS: If connected to a Hadoop cluster, you can select file(s) or folders to import. See *HDFS Browser*.

Hive: If connected to a Hive instance, you can load datasets from individual tables within the set of Hive databases. See *Hive Connections*.

S3: If connected to an S3 instance, you can browse your S3 buckets to select source files.

Tip: For HDFS and S3, you can select folders, which selects each file within the directory as a separate dataset.

See *External S3 Connections*.

Redshift: If connected to an S3 data warehouse, you can import source from the connected database. See *Amazon Redshift Connections*.

WASB: If enabled, you can import data into your Azure deployment from WASB.

ADL: If enabled, you can import data into your Azure deployment from ADLS Gen1.

ADLS Gen2: If enabled, you can import data into your Azure deployment from ADLS Gen1.

Databases: If connected to a relational datastore, you can load tables or views from your database. See *Database Browser*.

NOTE: For long-loading relational sources, you can monitor progress through each stage of ingestion. After these sources are ingested, subsequent steps to import and wrangle the data may be faster.

For more information, see *Configure JDBC Ingestion*.

For more information, see *Overview of Job Monitoring*.

For more information on the supported input formats, see *Supported File Formats*.

New/Edit: Click to create or edit a connection. By default, the displayed connections support import.

Search: Enter a search term to locate a specific connection.

NOTE: This feature may be disabled in your environment. For more information, contact your Trifacta administrator.

See *Create Connection Window*.

2. Add datasets

When you have found your source directory or file:

- You can hover over the name of a file to preview its contents.

NOTE: Preview may not be available for some sources, such as Parquet.

- Click the Plus icon next to the directory or filename to add it as a dataset.

Tip: You can import multiple datasets at the same time. See below.

- **Excel files:** Click the Plus icon next to the parent workbook to add all of the worksheets as a single dataset, or you can add individual sheets as individual datasets.

- If custom SQL query is enabled, you can click **Create Dataset with SQL** to enter a customized SQL statement to pre-filter the table within the database to include only the rows and columns of interest.

Through this interface, it is possible to enter SQL statements that can delete data, change table schemas, or otherwise corrupt the targeted database. Please use this feature with caution.

For more information, see *Create Dataset with SQL*.

If parameterization has been enabled, you can apply parameters to the source paths of your datasets to capture a wider set of sources. Click **Create Dataset with Parameters**. See *Create Dataset with Parameters*. To show hidden files or folders, select **Show hidden**.

NOTE: Hidden folder names begin with a dot (.) or underscore (_). In general, these folders are hidden for a reason. File structures may change without notice.

Tip: If you have run a job with profiling enabled, you can import your profile files as datasets and then publish them to other datastores, such as BigQuery, for additional analysis. These files are stored in the `.profiler` folder beneath your job results folder in `jobrun`. For more information on these files, see *Overview of Visual Profiling*.

3. Configure selections

When a dataset has been selected, the following fields appear on the right side of the screen. Modify as needed:

- **Dataset Name:** This name appears in the interface.
- **Dataset Description:** You may add an optional description that provides additional detail about the dataset. This information is visible in some areas of the interface.

Tip: Click the Eye icon to inspect the contents of the dataset prior to importing.

Tip: You can select a single dataset or multiple datasets for import.

Edit settings

You can edit any additional or optional settings for an individual dataset. Perform the following:

Steps:

1. Click **Edit Settings** from the card for an individual dataset in the right panel. The dialog box is displayed.
2. In the dialog box, select the required options and modify the settings.
 - **File Import Settings:** For more information, see *File Import Settings*.
 - **Table Import Settings:** For more information, see *Table Import Settings*.

4. Import selections

You can import one or more datasets. Continue selecting sources, and additional dataset cards are added to the right panel.

NOTE: If you are importing from multiple files at the same time, the files are not necessarily read in a regular or predictable order.

NOTE: When you import a dataset with parameters from multiple files, only the first matching file is displayed in the right panel.

In the right panel, you can see a preview of each dataset and make changes as needed.

×

5 New Datasets

Clear All

POS-r02.txt

×

Add a Description

# Store_Nbr	# Item_Nbr	#
101	322000	20
101	323000	20
101	325000	20
101	326000	20
101	327000	20

Edit settings

POS-r01.txt

×

Add a Description

# Store_Nbr	# Item_Nbr	#
1	381000	20
2	325000	20
2	325000	20
2	402000	20

☐ Add to new flow

Cancel

Continue

Figure: Importing Multiple Datasets


- To import the selected datasets, click **Continue**.

- To remove a dataset from import, click the X in the dataset card.

Database Browser

Contents:

- *Browse Databases*
- *Search List*
- *Preview Table Data*
- *Create Dataset with SQL*

**Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

The database browser enables you to interact with databases that are connected to the Designer Cloud powered by Trifacta® platform .

The database browser appears when:

- You select one of the database tabs to create an imported dataset. See *Import Data Page*.
- You choose a database connection through which to write job results.

For more information about interacting with databases through the product, see *Using Databases*.

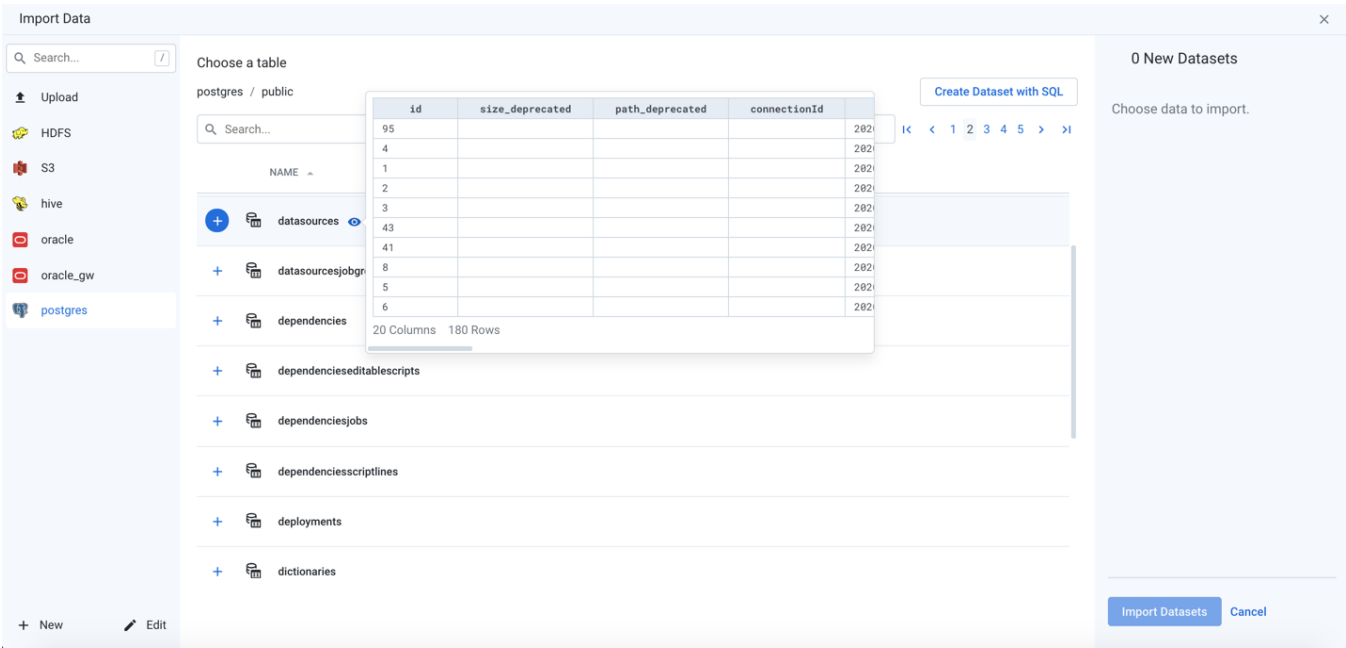






Figure: Database Browser

Browse Databases

Use the links and icons to browse databases and their tables and views.

NOTE: Avoid using the Back button on your browser, which exits the browser without applying changes to your configuration.

--	--	--

Identifier	Type	Description
	Database	Click these links to open a database to reveal its tables and views.
	Schema	(Postgres only) Click a schema link to display the tables and views that use the schema.
	Table	<p>Click the Plus icon to select this table.</p> <p>To preview its data, hover over the name of the table, and then click the Eye icon.</p> <div> <p>Tip: Sizes and update timestamps are calculated and displayed next to tables. They are not displayed next to databases.</p> <p>NOTE: Column count information is not available for nested tables.</p> </div>
	View	<p>Click the Plus icon to select this view as your source.</p> <p>To preview its data, click the Eye icon next to the view name.</p> <div> <p>NOTE: Previewing complex views may impact performance.</p> </div>
	Page navigation	<p>Use these links to navigate between pages of databases and tables and views.</p> <div> <p>NOTE: In some cases, subsequent pages of tables and views may be blank, and counts of tables and views may not match displayed figures. This is a known issue.</p> </div>
postgres / public	Breadcrumb	Click the links in the breadcrumb trail to navigate.

Search List

To filter the list, enter a string in the Search box. The filter is applied as you type and matches anywhere in the name of a currently displayed database, table, or view name.

Preview Table Data

Database tables are displayed by name only. To preview the data in the table, click the Eye icon next to the name of the table.

Tip: Table previews include available metadata information, such as column headers and column and row counts.

NOTE: Depending on the database type, rows may not be displayed in a specific order.

Create Dataset with SQL

As needed, you can pre-filter a selected table or view inside the database prior to import. By entering a custom SQL statement, you can remove unnecessary data from the dataset that is extracted from the database, which enables faster and more meaningful imports of your database data. See *Create Dataset with SQL*.

File System Browser

In the Designer Cloud powered by Trifacta® platform , the file system browser lets you browse, select, and filter the sources that you can access through the datastore to which you are connected. You also use the browser to select targets for publishing job results.

Interactions with the connected file system may be determined base on:

- user permissions to specific directories
- features enabled in the product
- any impersonation or kerberos restrictions

For more information, please contact your administrator.

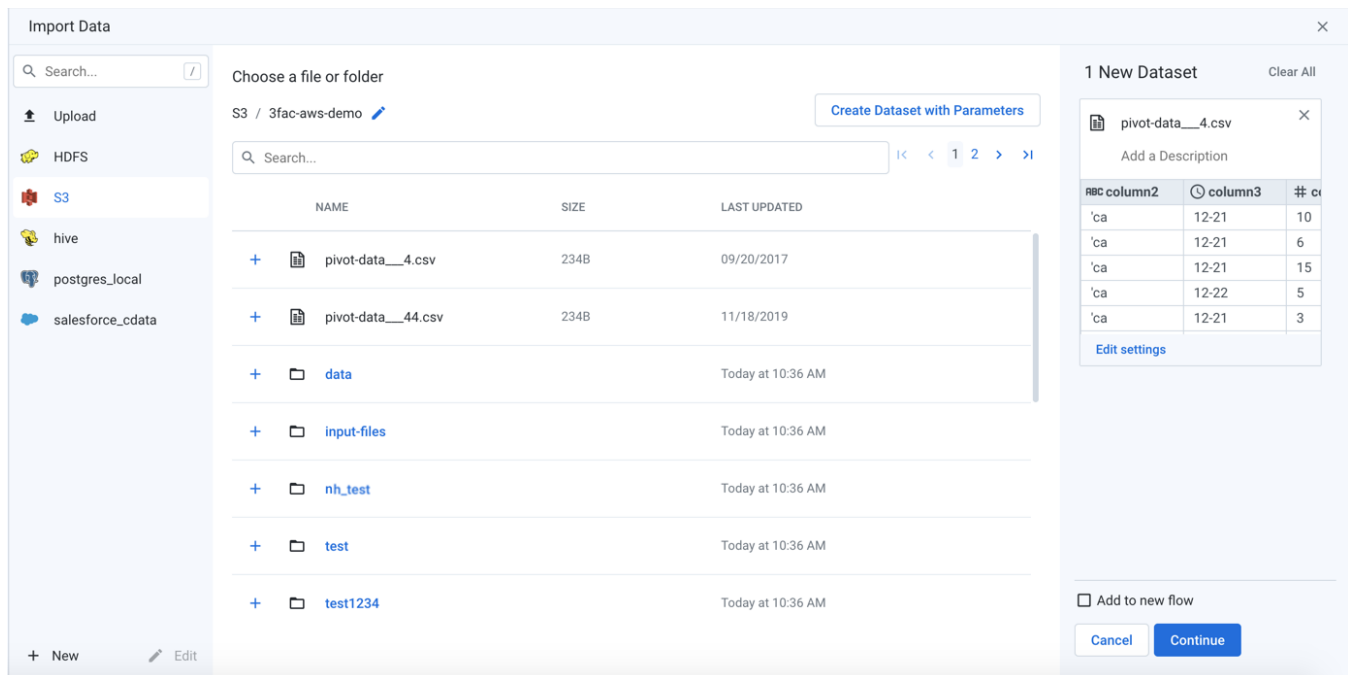


Figure: File System Browser

Through the file system browser, you navigate folders and select files through an easy-to-use interface. At a technical level, these objects are typically distributed across multiple servers and may be represented as part files of the whole virtual file.

Browse:




NOTE: Avoid using the Back button on your web browser, which exits the file system browser without applying changes to your configuration.

Use the links and icons to browse for files and folders in the file system tree structure.

NOTE: If you do not have the appropriate permissions, you may not be able to browse all of the folders of the directory. However, you may be able to paste in the full path to your location to gain access.

To display hidden files or folders, select **Show hidden**.

Tip: If the displayed file system is the base storage layer, then the path to your output home directory should be available through the browser.

Identifier	Type	Description
	Bucket	(not always present) In some file systems, the top-level browsing object is called a bucket . NOTE: You cannot add an entire bucket as a source of data for your datasets.
	Folder	<ul style="list-style-type: none">Click the Plus icon to select all readable files in this folder.Click the text link to open the folder and browse further. You must have the appropriate permissions in your account. Tip: When you open a new folder, a reference to it is added to the Path value. You can modify the path value manually, which may be a faster way to navigate up a deep directory structure. Tip: Sizes are displayed next to files. They are not displayed next to folders.
	File	Click the Plus icon to select this file. The Last Updated column contains information only for files. It is not available for directories.

Specify Path:

In the browser, you can specify an explicit path to resources. Click the Pencil icon, paste the path value, and click **Go**.

For example, if your home input directory is the following:

```
/mydir/input/username@example.com
```

You should paste the following in the Path textbox:

```
<bucketname>/mydir/input/username@example.com
```

Tip: You can retrieve your home directory from your profile.

Search Files:

To display a subset of files, enter a string in the Search box. The filter is applied as you type and matches anywhere in the name of a currently displayed file or folder.

NOTE: If you have a folder and file with the same name, search may only retrieve the file. You can still navigate to locate the folder.

HDFS Browser

The HDFS browser enables you to browse, select, and filter the files to which you have access in the Hadoop cluster to which Designer Cloud Powered by Trifacta® Enterprise Edition is connected.

The HDFS browser appears when you create a dataset in HDFS or in the HDFS tab when you import a dataset. See *Import Data Page*.

NOTE: Interactions with HDFS are determined by user permissions and features enabled in the Designer Cloud powered by Trifacta platform . For more information, see *Using HDFS*.

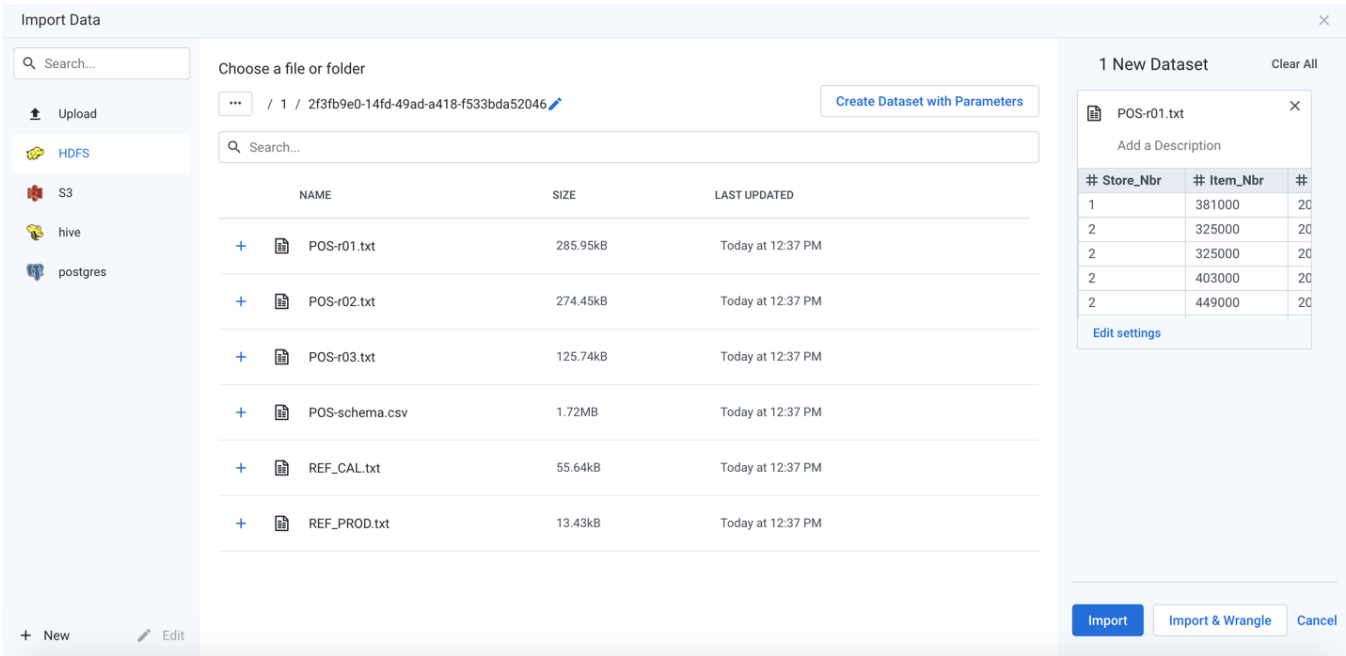



Figure: HDFS Browser

Browse HDFS:

Use the links and icons to browse for files and folders in the HDFS tree structure.

NOTE: Avoid using the Back button on your browser, which exits the HDFS browser without applying changes to your configuration.

To display hidden files or folders, select **Show hidden**.

Identifier	Type	Description
	Folder	<div><ul style="list-style-type: none">Click the Plus icon to select all readable files in this folder.Click the text link to open the folder and browse further.</div> <div>Tip: When you open a new folder, a reference to it is added to the Path value. You can modify the path value manually, which may be a faster way to navigate up a deep directory structure.</div> <div>Tip: Sizes are displayed next to files. They are not displayed next to folders.</div>



File

Click the Plus icon to select this file.

Specify HDFS Path:

In the HDFS browser, you can specify an explicit path to resources. Click the Pencil icon, paste the path value, and click **Go**.

```
/trifacta/input/username@example.com
```

You should paste the following in the Path textbox:

```
HDFS/trifacta/input/username@example.com
```

Tip: You can retrieve your home directory from your profile. See *Storage Config Page*.

Filter Files:

To display a subset of files, enter a string in the Search box. The filter is applied as you type and matches anywhere in the name of a currently displayed file or folder.

S3 Browser

In the Designer Cloud powered by Trifacta® platform , the S3 browser lets you browse, select, and filter the sources that you can access through S3. You also use the browser to select targets for publishing job results.

NOTE: Interactions with S3 are determined by user permissions and features enabled in the Designer Cloud powered by Trifacta platform . For more information, see *Using S3*.

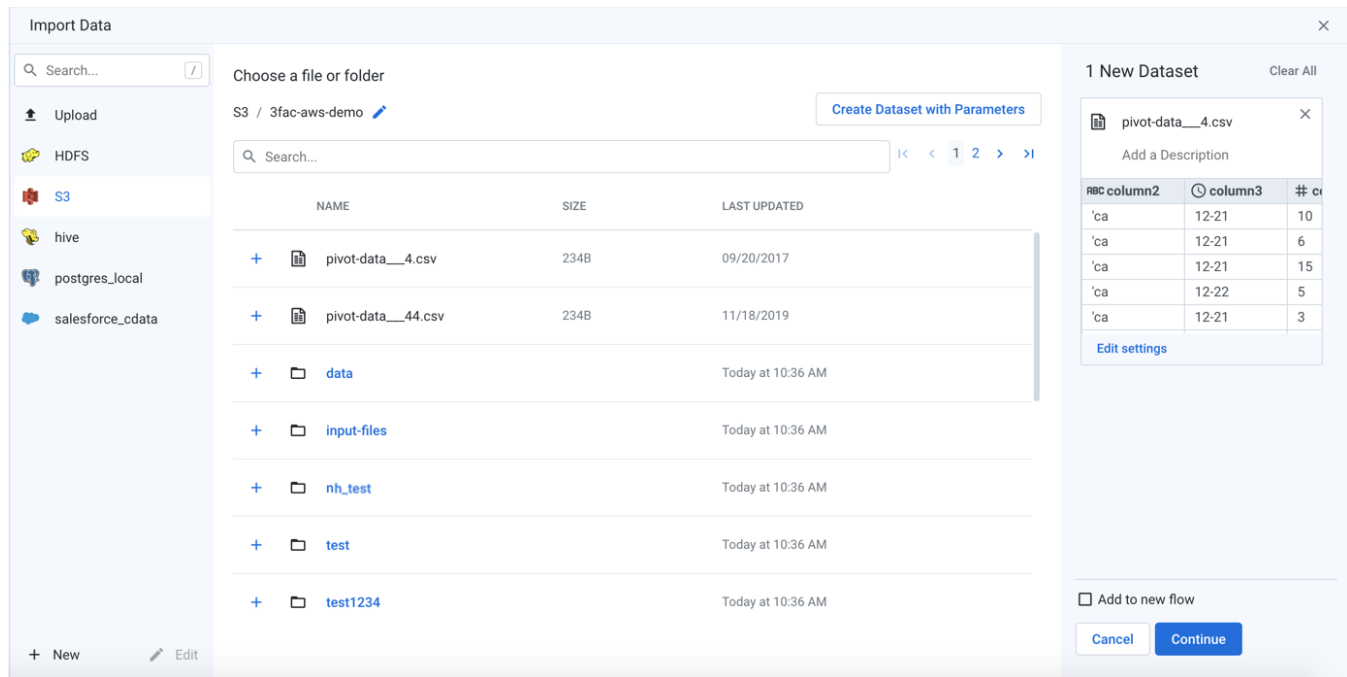


Figure: S3 Browser

Browse S3:

Use the links and icons to browse buckets for files and folders in the S3 tree structure.

NOTE: Permission to browse your buckets is determined by the permissions associated with your S3 credentials. If you do not have the appropriate permissions, you may not be able to browse the bucket. However, you may be able to paste in the full path to your location to gain access.




To display hidden files or folders, select **Show hidden**.

Tip: If S3 is the base storage layer, then the path to your output home directory should be available through the S3 browser. For more information on this path, see *Storage Config Page*.

- The Last Updated column contains information only for files. It is not available for directories.

NOTE: Avoid using the Back button on your browser, which exits the S3 browser without applying changes to your configuration.

Identifier	Type	Description
------------	------	-------------

	Bucket	Indicates an S3 bucket. NOTE: You cannot add an entire S3 bucket as a source of data for your datasets.
	Folder	<ul style="list-style-type: none"> Click the Plus icon to select all readable files in this folder. Click the text link to open the folder and browse further. You must have the appropriate permissions in your S3 account. <p>Tip: When you open a new folder, a reference to it is added to the Path value. You can modify the path value manually, which may be a faster way to navigate up a deep directory structure.</p> <p>Tip: Sizes are displayed next to files. They are not displayed next to folders.</p>
	File	Click the Plus icon to select this file.

Specify S3 Path:

In the S3 browser, you can specify an explicit path to resources. Click the Pencil icon, paste the path value, and click **Go**.

For example, if your home input directory is the following:

```
/mydir/input/username@example.com
```

You should paste the following in the Path textbox:

```
<bucketname>/mydir/input/username@example.com
```

NOTE: The name of the bucket (<bucketname>) must appear at the beginning of the path. Do not add a backslash (/) as a prefix.

Tip: You can retrieve your home directory from your profile. See *Storage Config Page*.

Search Files:

To display a subset of files, enter a string in the Search box. The filter is applied as you type and matches anywhere in the name of a currently displayed file or folder.

NOTE: If you have a folder and file with the same name in S3, search only retrieves the file. You can still navigate to locate the folder.

File Import Settings

Contents:

- *Per-file encoding*
 - *Detect structure*
 - *Remove special characters from column names*
 - *Selecting column headers*
-

When you edit settings on a selected file in the Import Data page, the following settings are displayed.

You can edit any additional or optional settings for an individual dataset. Perform the following:

1. Click **Edit Settings** from the card for an individual dataset in the right panel. The dialog box is displayed.
2. In the dialog box, select the required options and modify the settings.

Per-file encoding

By default, the Designer Cloud powered by Trifacta® platform applies a specified encoding type on the imported file. In some cases, the data preview panel may contain garbled data, due to a mismatch in encodings. In the Data Preview dialog, you can select a different encoding for the file. When the correct encoding is selected, the preview displays the data as expected.

NOTE: Assessing the file encoding type based on parsing an input file is not an accurate method. Instead, the Designer Cloud powered by Trifacta platform assumes that the file is encoded in the default encoding. If it is not, you should change the encoding type for the file.

NOTE: In some cases, imported files are not properly parsed due to issues with encryption types or encryption keys in the source datastore. For more information, please contact your datastore administrator.

Detect structure

By default, the Designer Cloud powered by Trifacta platform attempts to interpret the structure of your data during import. This structuring attempts to apply an initial tabular structure to the dataset.

- Unless you have specific problems with the initial structure, you should leave the Detect structure setting enabled.

Tip: When Detect structure is enabled for an imported dataset, the structure of the file can be checked for changes during job execution.

- When detecting structure is disabled, imported datasets whose schema has not been detected are labeled, **unstructured datasets**.
- When you create a recipe using the imported dataset:
 - When Detect structure is enabled, an initial set of parsing steps is applied to the data as the first steps of your recipe. These steps are hidden from view.

- When Detect structure is disabled, the initial parsing steps are inserted as the first steps of your recipe, where you can modify them to fit the structure of the data.
- For more information, see *Recipe Panel*.
- For more information, see *Initial Parsing Steps*.

Remove special characters from column names

When selected, characters that are not alphanumeric or underscores are stripped, and space characters are converted to underscores.

Selecting column headers

You can apply the column headers to your datasets during import. Select the required option from the drop-down list:

- **Infer header:** (default) When selected, the Designer Cloud application infers the header based on the data in the import.
- **Use first row as header:** When selected, the first row is used as the column headers.
- **No header:** When selected, the inference is ignored and column headers are defined using generic names with no headers.

If replacing a file:

- If you replace a dataset and select the **Use first row as header** option, then the existing header row labels are updated with the new headers.
- Pre-existing transformation steps may be broken if the headers are changed by a replaced file.

Table Import Settings

When you edit settings for a selected table in the Import Data page, the following settings are displayed.

You can edit any additional or optional settings for an individual dataset. Perform the following:

1. Click **Edit Settings** from the card for an individual dataset in the right panel. The dialog box is displayed.
2. In the dialog box, select the required options and modify the settings.

Infer column data types

You can choose whether or not to apply the Designer Cloud powered by Trifacta® platform type inference to table data imported from a database.

- In the preview panel, you can see the data type that is to be applied after the dataset is imported. This data type may change depending on whether column data type inference is enabled or disabled for the dataset.
- To enable the Designer Cloud powered by Trifacta platform type inference, select the Infer column data types checkbox.

Tip: To see the effects of the Designer Cloud powered by Trifacta platform type inference, you can toggle the checkbox and review data type listed at the top of individual columns. To override an individual column's data type, click the data type name and select a new value.

For more information, see *Disable Type Inference*.

You can configure the default use of type inference at the individual connection level. For more information, see *Create Connection Window*.

References Page

A reference dataset is a link to a recipe's output, which can be added to other flows.

NOTE: A reference dataset must be created from the context menu of a specific recipe. When added to another flow, a reference dataset is a read-only object.

For more information on creating reference datasets, see *View for Reference Datasets*.

References

All Owned by me Shared with me






Name	Owner	In flows	Source	Last updated ▾
 Advertising_Clickstream_wal...	anagarajan	2	Untitled Flow – 32	Yesterday at 1:53 PM
 Dataset-ProductNames	anagarajan	1	validation_test	Yesterday at 1:46 PM
 USDA_Farmers_Market_2014	Steve Olson	2	Farmers Market	Last Monday at 6:23 PM
 scraped_metadatarresults	anagarajan	1	JSONwithSteps	07/15/2021
 test_recipe	anagarajan	2	testing	05/25/2021

Figure: References Page

To create a reference dataset, click the plus icon next to any recipe you want and select **Create Reference Dataset**. A reference dataset is created from the recipe asset.

You can filter the reference datasets based on the ownership of the datasets.

- **All:** All reference datasets which you have access.
- **Owned by me:** All reference datasets that you own.
- **Shared with me:** All reference datasets that have been shared with you.

Columns:

- **Name:** Name of the reference dataset.
- **Owner:** Owner of the reference dataset.
- **In flows:** Count of flows where the reference dataset is used.
- **Source:** Flow from which the reference dataset was created.
- **Last Updated:** Timestamp when the reference dataset was modified.

Actions:

- **Browse:** If displayed, use the page browsing controls to explore the available objects.
- **Search:** To search object names, enter a string in the search bar. Results are highlighted immediately in the page.
- **Sort:** Click a column header to sort the display by the column's entries.

Context menu options:

NOTE: The context menu options vary based on the ownership of the datasets.

- **Preview:** Inspect a preview of the dataset.

NOTE: Preview is not available for binary format sources.

- **Add to Flow:** Add the dataset to a new or existing flow.
- **Edit name and description:** Change the name and description of the dataset.
- **Delete Reference Dataset:** Delete the dataset.

Deleting a reference dataset causes all references to it to be broken in the flows where it had been added. These broken references should be fixed by swapping in new sources.

Macros Page

In the Macros page, you can review and manage the macros to which you have access.

A **macro** is a saved sequence of one or more recipe steps that can be reused in other recipes. Values in your macros can be parameterized. For more information, see *Overview of Macros*.




Macros			Find Macros...
Name	Used in	Last Updated ▾	
 InitialCleanv3 Initial steps cleanup v3	1 Flow • 1 Recipe	Today at 5:36 PM	
 InitialCleanV2 Initial cleanup steps	0 Flows • 0 Recipes	Today at 5:34 PM	
 InitialClean Initial recipe cleanup steps	0 Flows • 0 Recipes	Today at 5:31 PM	

Figure: Macros Page

To review specifics about the macro, click its name. See *Macro Details Page*.

Columns:

- **Name:** Name of the macro.
- **Used in:** Count of flows and recipes in which the macro is used.
- **Last Updated:** Timestamp for when the macro was last modified.

Actions:

- **Import Macro:** Click to import a macro that has been exported to your local desktop. For more information, see *Import Macro*.
- **Get Macros:** Explore and download macros from Wrangle Exchange.
- **Search:** Enter a string in the search box. The list of macros is updated in real-time.
- **Sort:** Click the caret next to any column head to sort the list based on the column.
- **Context Menu:** See below.

Context menu options:

- **Edit:** Modify the name and description for the macro. You can also modify the name, description, and default values for the macro's inputs.
- **Inspect:** Review the recipe steps in the macro.
- **Export:** Export the macro to your local desktop. For more information, see *Export Macro*.
- **Replace:** Replace an existing macro definition with a macro that you have exported to your local desktop.

NOTE: If your imported macro contains macro inputs that are not in the macro that you are replacing, the existing instances of the replaced macro contain a broken step where the macro is referenced but has no data. These references must be fixed in each macro instance.

Tip: If you must add more macro inputs or steps to a macro that you have imported, you must convert the macro to steps, modify them, and then perform a replacement. For more information, see *Create or Replace Macro*.

- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user.

NOTE: After a macro is transferred, the original owner no longer has collaborator access to the macro, since macros cannot be shared.

See *Transfer Asset Ownership*.

- **Delete:** Delete the macro.

NOTE: When a macro is deleted, in any recipe that references it, the macro's steps are expanded into regular recipe steps. Any macro inputs are applied as static values in the expanded recipe steps.

Deleting a macro cannot be undone.

Macro Details Page

Through the Macro Details Page, you can review details about an individual macro. In the Macros page, click the name of the macro to review.

Actions:

To modify the macro name and description, click **Edit**.

Tip: You can also modify the name, description, and default values for the macro's inputs.

Context menu:

- **Export:** Export the macro to your local desktop. See *Export Macro*.
- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user.

NOTE: After a macro is transferred, the original owner no longer has collaborator access to the macro, since macros cannot be shared.

See *Transfer Asset Ownership*.

- **Delete:** Delete the macro.

NOTE: When a macro is deleted, in any recipe that references it, the macro's steps are expanded into regular recipe steps. Any macro inputs are applied as static values in the expanded recipe steps.

Deleting a macro cannot be undone.

Overview Tab

In the Overview tab, you can review the steps in the macro.

Macros

InitialCleanv3

Initial steps cleanup v3

Overview

Used in

Macro steps

1

rename type: findAndReplace col: * on: '' with: '' matchAll: true

2

derive type: single value: \$sourcerownumber as: \$rowindex

3

move col: \$rowindex position: after after: \$Current_HO_Retail

Details

Created

Today at 5:36 PM

Last updated

Today at 5:36 PM

Edit

...

Figure: Macro Detail Page - Overview tab

- Steps are displayed in raw Wrangle .
- You can review creation and update timestamps.

Used In Tab

In the Used In tab, you can review all of the recipes and flows where the macro is referenced.



<div>Macros</div> <div><div>InitialCleanv3</div><div>Initial steps cleanup v3</div></div> <div>Edit...</div>		
<div>OverviewUsed in</div>		
Name	Flow	Last Updated ▾
<div>POS-r01</div>	POS-r01 Flow	Today at 5:36 PM

Figure: Macro Detail Page - Used In tab

- Click the name of the recipe or flow to open the flow.

Library Tasks

Through the Library, you can add and manage your assets in the Designer Cloud powered by Trifacta® platform .

Import a File

You can import one or more files into the Designer Cloud powered by Trifacta® platform .

NOTE: When you import a file, the data is not stored in the Designer Cloud powered by Trifacta platform . What you create is an **imported dataset**, which is simply a reference to the source of the data. The Designer Cloud powered by Trifacta platform never stores or modifies source data.

NOTE: Before you begin, you should review information on file formats supported for import, which cause your files to fail to import or to be properly ingested and formatted. For more information, see *Supported File Formats*.

Steps:

- From the menubar, click **Library**.
- In the Library page, click **Import Data**.
- From the left sidebar in the Import Data page, select the connection where your data is located.
 - You must have read permissions on any directory and file that you wish to import.
 - **Upload:** Navigate your local desktop to select the file or files that you wish to upload.

Tip: You can select multiple files in the same directory for uploading at the same time.

- **File-based datastore:** If you are uploading from a file-based backend datastore, navigate the available directories to locate your file.
 - **Microsoft Excel:** If you are importing an Excel file that contains multiple worksheets, you must select the worksheets to include as part of your import.
 - **Dataset with Parameters:** If you are importing multiple files with similar filenames, you can import them as part of the same dataset using parameters or variables. In this manner, you create a single imported dataset, which automatically includes any new files that appear in the directory and that follow the same file naming pattern.
- Some aspects of the import process can be modified. In the right panel, click **Edit Settings** for a file that you have imported.
 - **Detect structure:** By default, the application attempts to detect the file structure to convert it into a tabular format. Disable this setting only if you are experiencing issues with automated detection.
 - If your file uses a different file encoding than the default encoding, you can change it for the file during the import process.
- When you are ready to complete the import process:

Tip: If present, you can click the **Add to new flow** checkbox, which adds the imported datasets to an *Untitled* flow.

- Your files are available as imported datasets.

For more information, see *Import Data Page*.

Change File Encoding

Files are imported based on the default file encoding for the Designer Cloud powered by Trifacta® platform .

The default file encoding can be configured. For more information, see *Configure Global File Encoding Type*.

As needed, you can override the default file encoding during the importing of individual datasets.

NOTE: All output files are written in UTF-8 encoding.

Tip: If you have already imported the dataset and need to change this setting, you can re-import the source and change the settings. References to this imported dataset must be updated.

Steps:

1. After you have selected or specified the file to import in the Import Data page, click **Edit Settings** for the dataset card in the right panel.
2. From the drop-down, select the preferred encoding to apply to this specific file.
3. Continue the import process.

Remove Initial Structure

When you import a dataset from a file, the Designer Cloud powered by Trifacta® platform attempts to detect the structure of the file and to apply an initial set of parsing steps to the data to render it in tabular form for display in the Transformer page. For example, JSON files may be turned into a table of data as long as the structure of the data supports this structuring.

NOTE: Initial parsing steps are applied only to file-based sources of data.

These steps vary based on the file format of data that is being imported. Depending on the dataset, you may need to modify these steps or rebuild them altogether. You can use the following steps to prevent the Designer Cloud powered by Trifacta platform from detecting the structure and automatically hiding these steps.

Tip: You should allow the product to detect the structure first. If it does not detect the structure well, you can experiment with disabling it and rebuilding the steps to render your data in tabular format.

Tip: If you have already imported the dataset and need to change this setting, you can re-import the source and change the settings. Wherever the previously imported version of this dataset is referenced, you must change the reference to use this newly imported dataset.

NOTE: When the steps are completed, the initial parsing steps are listed in any recipe that you create from the imported dataset. If you wish to remove them altogether, you can delete them from the recipe.

Import Unstructured Dataset

1. After you have selected or specified the file to import in the Import Data page, click **Edit Settings** for the dataset card in the right panel.
2. Deselect the Detect Structure checkbox. For more information, see *File Import Settings*.
3. Continue the import process by adding the dataset.
4. When the imported dataset is added, it is listed as an **unstructured dataset**.

Use Unstructured Dataset

1. Import the unstructured dataset in Flow View.
2. Select the unstructured dataset and click **Add new recipe**.
3. When you select the recipe, the initial parsing steps are listed in the right panel.
4. When the dataset is loaded into the Transformer page, you can modify these steps to improve the parsing or delete them altogether.

NOTE: Any step that breaks up the data into individual rows into individual rows must be the first step in the recipe. To create, enter **Break into rows** in the Search panel.

Import a Table

You can import one or more tables from relational sources into the Designer Cloud powered by Trifacta® platform .

NOTE: When you import a file, the data is not stored in the Designer Cloud powered by Trifacta platform . What you create is an **imported dataset**, which is simply a reference to the source of the data. The Designer Cloud powered by Trifacta platform never stores or modifies source data.

Steps:

- From the menubar, click **Library**.
- In the Library page, click **Import Data**.
- From the left sidebar in the Import Data page, select the connection to the relational datastore where your data is located.
- Browse the relational datastore to locate the table that you wish to import.

Tip: You can select multiple tables from the database and add it to a flow.

NOTE: You must have read permissions on any database and table that you wish to import.

- **Create Dataset with SQL:** You can apply a custom SQL statement to import from a database. For more information, see *Create Dataset with SQL*.
- In the right panel, click **Edit Settings** to modify how the table or tables are imported.
 - The application's data types are applied to the table's columns during the import process. If needed, you can disable type inference, so that the data types of the original source are preserved, if possible, during import. For more information, see *Disable Type Inference*.
- When you are ready to complete the import process:

Tip: If present, you can click the **Add to new flow** checkbox, which adds the imported datasets to an **Untitled** flow in Flow View.

- Your tables are available as imported datasets.

For more information, see *Import Data Page*.

Disable Type Inference

When the Designer Cloud powered by Trifacta® platform creates an imported dataset from a schematized source, the product applies its own type inferencing to the columns of the imported data. Type inferencing may be reapplied during some operations, such as the creation of samples or when data reshaping transformations are applied in the Transformer page.

If preferred, you can disable this type inferencing on the columns of your imported dataset. When the data is imported, the original types from the source system remain. Any types that do not have a corresponding match with the Trifacta data types must be manually typed in the application.

Methods of disabling:

Column data typing is applied to schematized sources in one of three ways:

1. Globally
2. Per-connection type inference settings override the global setting.
3. Per-file type inference settings override both global and per-connection settings.

For more information on applying global or per-connection type inference settings, see *Configure Type Inference*.

You can use the following steps to disable type inference applied to a specific file during the import process.

Tip: For imported datasets from relational sources, you can identify in Flow View whether type inferencing has been applied to the dataset. When the dataset is selected in Flow View, locate the Type Inference entry in the right panel.

Tip: If you have already imported the dataset and need to change this setting, you can re-import the source and change the settings. In any locations where the previously imported version of this dataset is used, you can update references to use this new version. this newly imported version.

Steps:

1. After you have selected or specified the relational table to import in the Import Data page, click **Edit Settings** for the dataset card in the right panel.
2. Deselect the Infer column data types checkbox.
3. Continue the import process.

Create Dataset with Parameters

Contents:

- *From File System*
 - *Parameterize bucket names*
 - *From Relational Sources*
 - *Edit Parameter*
 - *Apply Parameter Overrides*
 - *Apply parameter overrides for your flow*
 - *Apply parameter overrides for your job*
 - *Delete Parameter*
-

This section provides an overview on how to parameterize relational sources and files while importing data into the Designer Cloud powered by Trifacta® platform .

For more information on parameterization of datasets and other types of parameters, see *Overview of Parameterization*.

From File System

When browsing for data on your default storage layer, you can choose to parameterize elements of the path. Through the Import Data page, you can select elements of the path, apply one of the supported parameter types and then create the dataset with parameters.

NOTE: When you import a file, the data is not stored in the Designer Cloud powered by Trifacta® platform . What you create is an imported dataset that is simply a reference to the source of the data. The Designer Cloud powered by Trifacta platform never stores or modifies source data.

When you create a dataset with parameters in the Designer Cloud powered by Trifacta platform , you can replace segments of the input path with parameters. Suppose you have the following files that you'd like to capture through a parameterized dataset:

```
//source/user/me/datasets/month01/2017-01-31-file.csv
//source/user/me/datasets/month02/2017-02-28-file.csv
//source/user/me/datasets/month03/2017-03-31-file.csv
//source/user/me/datasets/month04/2017-04-30-file.csv
//source/user/me/datasets/month05/2017-05-31-file.csv
//source/user/me/datasets/month06/2017-06-30-file.csv
//source/user/me/datasets/month07/2017-07-31-file.csv
//source/user/me/datasets/month08/2017-08-31-file.csv
//source/user/me/datasets/month09/2017-09-30-file.csv
//source/user/me/datasets/month10/2017-10-31-file.csv
//source/user/me/datasets/month11/2017-11-30-file.csv
//source/user/me/datasets/month12/2017-12-31-file.csv
```

A parameterized reference to all of these files would look something like:

```
//source/user/me/datasets/month##/YYYY-MM-DD-file.csv
```

Through the application, you can specify the parameters to match all values for:

- **##** - You can use a wildcard or (better) a pattern to replace these values.
- **YYYY-MM-DD** - A formatted Datetime parameter can replace these values.

For more information, see *Parameterize Files for Import*.

Parameterize bucket names

You can create environment parameters for your bucket names.

From Relational Sources

You can create datasets from a relational source by applying parameters to the custom SQL that pulls the data from the source. During import of database tables through relational connections, you can apply parameters to the SQL query that you use to define the imported dataset. In some scenarios, you may need to define the table to import using a variable parameter or to parameterize the time value associated with a table name. Using parameters, you can define the tables, columns, and conditions of the query that you use to bring in data from a relational database.

For more information, see *Parameterize Tables for Import*.

Edit Parameter

After you have created your dataset with parameters, you can edit the parameter as needed.

Steps:

1. In the left nav bar, select **Library**.
2. In the Library page, locate the dataset. From its context menu, select either of the following:
 - a. **Files:** Select **Edit parameters**. In the Edit Dataset with Parameters, click the parameter to modify its definition.
 - b. **Tables:** Click **Edit Custom SQL**. In the Custom SQL window, you can modify the SQL statement, including any parameters in it. For more information, see *Create Dataset with SQL*.

Apply Parameter Overrides

After you have created a parameterized dataset, you can apply overrides to the default value. These override values can be applied in the following order of precedence.

- **Job:** When you choose to execute a job, you can set a new value for the parameter, which is applied for the specified job only.
- **Flow:** If your imported dataset containing a parameter is added to a flow, you can define an override value for the dataset's parameter through Flow View. Whenever a job is executed on the imported dataset within the flow, the override value is applied to the dataset.

NOTE: If a job-level override is applied on top of a flow-level override, the job override value is applied to the job.

- **Default:** The default value for the parameter is used if no override is applied.

Apply parameter overrides for your flow

Steps:

1. Open the flow.
2. In Flow View, select the icon for your dataset with parameters.
3. From the context menu, select **Parameter**.
4. In the Manager Parameters dialog, click the Overrides tab.
5. Edit the required values, click **Save**.

For more information, see *Manage Parameters Dialog*.

Apply parameter overrides for your job

You can apply parameter overrides to your job.

Steps:

1. In Flow View, select the output that you wish to generate.
2. In the right context panel, click **Run Job**.
3. In the Run Job page, you can specify job-level overrides at the bottom of the screen.

For more information, see *Run Job Page*.

Delete Parameter

Steps:

1. In the Edit Dataset with Parameters screen, select the parameter that you wish to remove.

NOTE: Before you remove parameter, you may want to take note of the default value, which may need to be applied to the path or query after you remove the parameter.

2. In the popup, click **Delete**.
3. Save your changes.
4. The parameter is removed from the imported dataset definition.

Parameterize Files for Import

Contents:

- *Structuring Your Data*
 - *Steps*
 - *Add Datetime Parameter*
 - *Extend Datetime parameter*
 - *Add Variable*
 - *Parameterize bucket names*
 - *Add Pattern Parameter*
-

This section describes how to create datasets and replace segments by parameterizing the input paths to your data in the Designer Cloud powered by Trifacta® platform .

Structuring Your Data

Each file that is included as part of the dataset with parameters should have identical structures:

- Matching file formats
- Matching column order, naming, and data type
- Matching column headers. Each column in any row that is part of a column header in a dataset with parameters should have a valid value that is consistent with corresponding values across all files in the dataset.

NOTE: If your files have missing or empty values in rows that are used as headers, these rows may be treated as data rows during the import process, which may result in unexpected or missing column values.

- Within each column, the data format should be consistent.
 - For example, if the date formats change between files in the source system, you may not be able to manage the differences, and it is possible that data in the output may be missing.

NOTE: Avoid creating datasets with parameters where individual files or tables have differing schemas. Either import these sources separately and then correct in the application before performing a union on the datasets, or make corrections in the source application to standardize the schemas.

When working with datasets with parameters, it may be useful to do the following if you expect the underlying datasets to be less than 100% consistent with each other.

- Recreate the dataset with parameters, except deselect the Detect Structure option during the import step.
- If possible, collect a Random Sample using a full scan. This step attempts to gather data from multiple individual files, which may illuminate problems across the data.

Tip: If you suspect that there is a problem with a specific file or rows of data (e.g. from a specific date), you can create a static dataset from the file in question.

Steps

NOTE: Matching file path patterns in a large directory can be slow. Where possible, avoid using multiple patterns to match a file pattern or scanning directories with a large number of files. To increase matching speed, avoid wildcards in top-level directories and be as specific as possible with your wildcards and patterns.

1. In the Import Data page, navigate your environment to locate one of the files or tables that you wish to parameterize.
2. Click **Create Dataset with Parameters**.

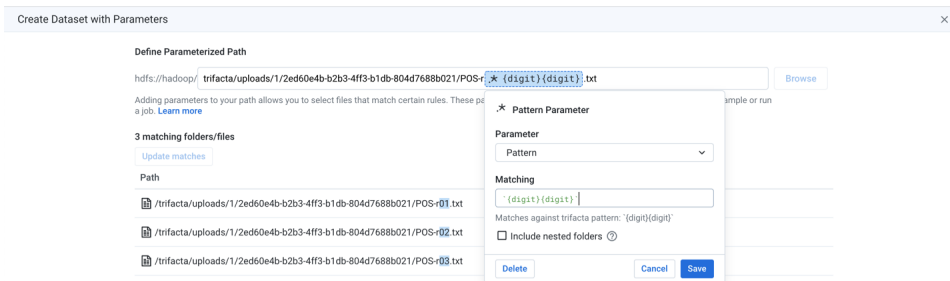


Figure: Create Dataset with Parameters

3. Within the Define Parameterized Path, select a segment of text. Then select one of the following options:

Tip: For best results when parameterizing directories in your file path, include the trailing slash (/) as part of your parameterized value.

- a. Add Datetime Parameter
 - b. Add Variable
 - c. Add Pattern Parameter - wildcards and patterns
 - d. If you need to navigate elsewhere, select **Browse**.
4. Specify the parameter. Click **Save**.
 5. Click **Update matches**. Verify that all of your preferred datasets are matching.

NOTE: If you are matching with more datasets than you wish, you should review your parameters.

6. Click **Create**.
7. The parameterized dataset is loaded.

Add Datetime Parameter

Datetime parameters require the following elements:

Format: You must specify the format of the matching date and/or time values using alphanumeric patterns. To review a list of example formats, click **Browse Date/Timestamp Patterns**.

You can also create custom formats using patterns. For example, the following regex pattern matches patterns like MM.DD.YYYY:

```
/[0-9][0-9]\.[0-9][0-9]\.[0-9][0-9][0-9][0-9]/
```

Date range: Use these controls to specify the range that matching dates must fall within.

NOTE: Date range parameters are case-insensitive.

Tip: Datetime parameters that you configure here are evaluated at the time of job execution. So, `now` refers to the time when the job is executed.

Time zone: The default time zone is the location of the host of the application. To change the current time zone, click **Change**.

For a list of supported time zone values, see *Supported Time Zone Values*.

Extend Datetime parameter

A parameterized dataset can support only one Datetime parameter. If you have multiple parts of the path that contain date information, you can create a Datetime element for each part.

Steps:

1. Within the Define Parameterized Path, select a segment of text for which to create the first part.
2. Create the Datetime parameter for this element. Remember to use the appropriate format for the part. For example, if you have highlighted a four-digit year for the part, the date format value should be: `YYYY`.
3. Then, select the second element and click the Extend Datetime Parameter icon.

The screenshot shows the 'Define Parameterized Path' interface. The path is `hdfs://trifacta/uploads/1/aa112099-4f6e-4925-93fa-d98677233965/`. Two segments are highlighted: `YYYY` (a four-digit year) and `04` (a two-digit month). A 'Datetime Parameter' dialog box is open, showing the 'Format' field set to `YYYY` and the 'Date range' field set to 'Date is last'. The dialog also shows an example of the path with the parameters applied: `/trifacta/uploads/1/aa112099-4f6e-4925-93fa-d98677233965/2018-04-01-orders.txt`.

Figure: Click the *Extend Datetime Parameter icon* to create additional parts to your *Datetime parameter*.

4. In the dialog, you can specify the date format of the second element of your Datetime parameter. Matches are made on the two elements, as well as any static text in between them.

Add Variable

A **variable** parameter is a key-value pair that can be inserted into the path.

- At execution time, the default value is applied, or you can choose to override the value.
- A variable can have an empty default value.

Name: The name of the variable is used to identify its purpose.

NOTE: If multiple datasets share the same variable name, they are treated as the same variable.

Tip: Type `env.` to see the environment parameters that can be applied. These parameters are available for use by each user in the environment.

Default Value: If the variable value is not overridden at execution time, this value is inserted in the variable location in the path.

NOTE: When you edit an imported dataset, if a variable is renamed, a new variable is created using the new name. Any override values assigned under the old variable name for the dataset must be re-applied. Instances of the variable and override values used in other imported datasets remain unchanged.

Parameterize bucket names

You can create environment parameters to specify your bucket names. An **environment parameter** is a variable name and String value that can be referenced by all users of the environment.

NOTE: A workspace administrator or project owner can create environment parameters.

Uses:

- Parameterized bucket names are very useful when you are moving assets between workspaces or projects. When the asset is imported into a new workspace, the environment parameter references the appropriate bucket name in the new workspace.
- If you change source buckets or move data to a new storage bucket, updating the paths to your objects can be as simple as changing the value of the environment parameter where your data is stored.

For example, suppose you have two environments: Dev and Prod. You can create an environment parameter called `env.sourceBucketName` to store the name of the bucket from which all data in the workspace or project is imported.

Environment Name	Source Bucket Name	Environment Parameter Value
Dev	MyCo_Dev	<code>\$env.sourceBucketName = 'MyCo_Dev'</code>
Prod	MyCo_Prod	<code>\$env.sourceBucketName = 'MyCo_Prod'</code>

For more information, see *Environment Parameters Page*.

Add Pattern Parameter

In the screen above, you can see an example of pattern-based parameterization. In this case, you are trying to parameterize the two digits after the value: `POS-r`.

Include nested folders

When you create a wildcard or pattern-based parameter, you have the option to scan any nested folders for matching sources.

- If disabled, the scan stops when the next slash (/) in the path is encountered. Folders are not matched.
- If enabled, the scan continues to any depth of folders.

NOTE: A high number of files and folders to scan can significantly increase the time required to load your dataset with parameters.

Example 1: all text files

Suppose your file and folder structure look like the following:

```
//source/user/me/datasets/thisfile.txt
//source/user/me/datasets/thatfile.txt
//source/user/me/datasets/anotherfile.csv
//source/user/me/datasets/detail/anestedfile.txt
//source/user/me/datasets/detail/anestedfile2.txt
//source/user/me/datasets/detail/anestedfile4.txt
//source/user/me/ahigherfile.txt
```

Since the filenames vary significantly, it may be easiest to create your pattern based on a wildcard. You create a wildcard parameter on the first file in the `//source/user/me/datasets` directory:

```
//source/user/me/datasets/*.txt
```

For the specified directory, the above pattern matches on any text file (.txt). In the example, it matches on the first two files but does not match on the CSV file.

When the Include nested folders checkbox is selected:

- The first two files are matched.
- The next three files inside a nested folder are matched.
- The last file (`ahigherfile.txt`) is not matched, since it is not inside a nested folder.

Example 2: pattern-based files

Suppose your file and folder structure look like the following:

```
//source/user/me/datasets/file01.csv
//source/user/me/datasets/file02.csv
//source/user/me/datasets/file03.csv
//source/user/me/datasets/detail/file04.csv
//source/user/me/datasets/detail/file05.csv
//source/user/me/datasets/detail/file06.csv
//source/user/me/file07.csv
```

You create a pattern parameter on the first file in the `//source/user/me/datasets` directory with the following pattern-based parameter:

```
`file{digit}+`
```

The above pattern matches on the word `file` and a sequence of one or more digits. For example, suppose `file100.csv` lands in the directory at some point in the future. This pattern would capture it.

In the above example, this pattern matches on the first three files, which are all in the same directory.

When the Include nested folders checkbox is selected:

- The first three files are matched.
- The next three files inside a nested folder are matched.
- The last file (`file07.csv`) is not matched, since it is not inside a nested folder.

Wildcard

The easiest way to is to add a wildcard: *

A **wildcard** can be any value of any length, including an empty string.

Tip: Wildcard matching is very broad. If you are using wildcards, you should constrain them to a very small part of the overall path. Some running environment may place limits on the number of files with which you can match.

Pattern - Regular expression

Instead of a wildcard match, you could specify a regular expression match. **Regular expressions** are a standardized means of expressing patterns.

Regular expressions are specified between forward slashes, as in the following:

```
/my_regular_expression/
```

NOTE: If regular expressions are poorly specified, they can create unexpected matches and results. Use them with care.

The following regular expression matches the same two sources in the previous screen:

```
/\[0-9]*\[0-9]*\
```

The above expression matches an underscore (`_`) followed by any number of digits, another underscore, and any number of digits.

Tip: In regular expressions, some characters have special meaning. To ensure that you are referencing the literal character, you can insert a backslash (`\`) before the character in question.

While the above matches the two sources, it also matches any of the following:

```
_2_1  
__1  
_1231231231231231235245234343_
```

These may not be proper matches. Instead, you can add some specificity to the expression to generate a better match:

```
/\[0-9]{13}\[0-9]{4}\
```

The above pattern matches an underscore, followed by exactly 13 digits, another underscore, and then another 4 digits. This pattern matches the above two sources exactly, without introducing the possibility of matching other numeric patterns.

Pattern - Trifacta pattern match

A Trifacta pattern is a platform-specific mechanism for specifying patterns, which is much simpler to use than regular expressions. These simple patterns can cover much of the same range of pattern expression as regular expressions without the same risks of expression and sometimes ugly syntax.

Trifacta patterns are specified between back-ticks, as in the following:

```
`my_pattern`
```

In the previous example, the following regular expression was used to match the proper set of files:

```
/\[0-9]{13}\_[0-9]{4}/
```

In a Trifacta pattern, the above can be expressed in a simpler format:

```
`\_ {digit}{13}\_ {digit}{4}`
```

This simpler syntax is easier to parse and performs the same match as the regular expression version.

Parameterize Tables for Import

Contents:

- *Import Parameterized Tables*
 - *Create a custom SQL dataset*
 - *Parameterize dataset with a variable*
 - *Parameterize dataset with a timestamp*
 - *Examples*
 - *Pre-filter rows from a table*
 - *Run a weekly job on daily tables*
 - *Parameterize entire query*
-

This section provides an overview on how to apply parameters to the tables that you import as datasets.

During import of database tables through relational connections, you can apply parameters to the SQL query that you use to define the imported dataset. In some scenarios, you may need to define the table to import using a variable parameter or to parameterize the time value associated with a table name. Using parameters, you can define the specific tables that you use to bring in data from a relational database.

Following are the type of parameters you can apply for relational sources:

- **Timestamps:** Inserts a formatted timestamp when creating a custom SQL query.
- **Variables:** Inserts a value for the variable. This variable has a default value that you assign.

NOTE: Pattern-based parameters are not supported for relational imports.

Import Parameterized Tables

While importing data, you parameterize relational tables by creating custom SQL statements to specify the dataset. By default, when you import a table from a relational source, the Designer Cloud powered by Trifacta® platform generates a `SELECT *` statement to import the entire table. The Custom SQL enables you to customize the query to pull the data from the source system.

The following are the prerequisites and procedures for parameterizing the relational sources table:

Prerequisites:

- A connection must be created for your target database.
 - Verify that you have access to a read-only or read-write set of connections.
 - For more information, see *Connect to Data*.

Limitations:

- The original table that is used as the default value in any custom SQL query remains the default table whenever the custom SQL dataset is used.
 - During job execution on a dataset of parameterized tables, if the parameterized SQL query returns data from a table from the default table, Designer Cloud powered by Trifacta platform expects that the schema of the new table matches the schema of the original default table.
 - If the schemas do not match, schema mismatch errors may be reported.
- Parameterizing the column project list in the `SELECT` portion of a custom SQL statement is not supported.
- Schema refresh of a parameterized dataset using custom SQL is not supported.

Create a custom SQL dataset

You can create a custom SQL dataset through the Import Data page.

Steps:

1. In the Designer Cloud application, click **Library** in the left nav bar.
2. In the Library page, click **Import Data**.
3. From the left side of the Import Data page, select the relational connection from which to import.
4. Depending on the type of relational connection, you may need to select the database or schema to browse.
5. Locate the tables to import. Take note of the table name or names.
6. Click **Create Dataset with SQL**. The Create Dataset with SQL window is displayed.

In this window, you specify the `SELECT` statement to retrieve the data from a table or tables that you specify.

NOTE: When specifying a SQL statement for your database, you are constructing a direct query of the database. You must use the syntax required by the database vendor.

For more information on creating datasets with SQL, see *Create Dataset with SQL*.

Parameterize dataset with a variable

A variable parameter enables you to insert variable into the query statement used to define your dataset. You can replace or highlight elements of the query to add parameters.

How to use variables:

- When a job is executed, the currently specified variable value is passed to the running environment. By default, the value that you specify as part of the dataset creation process is provided.
- You can override this value:
 - You can specify an override for a variable parameter through Flow View.
 - For any specific job run, you can specify an override value through the Run Job page.
- In this manner, you can specify the exact data that you wish to retrieve at the flow- or job-level.

Steps:

1. Create a custom dataset using SQL. For more information, see Create a Custom SQL Dataset above.
2. In the Create Dataset with SQL window, enter a `SELECT*` statement to retrieve data from the specified table. Click **Validate SQL** to verify that the query is properly specified.
3. Now, highlight the part of the query that you wish to parameterize. Click the Variable icon.

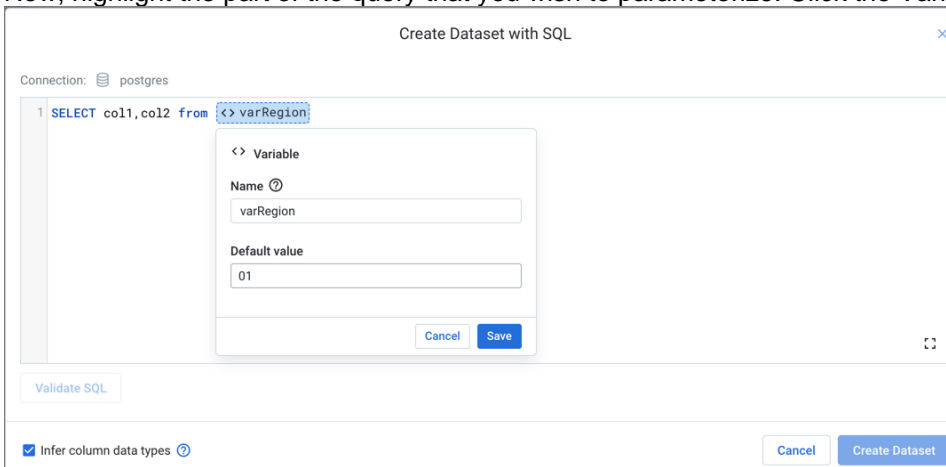


Figure: Define Variable Parameter

4. In the Variable dialog, enter the following details:

Tip: Type `env.` to see the environment parameters that can be applied. These parameters are available for use by each user in the environment.

- a. **Name:** Enter a display name for the variable.
 - b. **Default value :** Enter a default value for the parameter.
5. Click **Save** to save the parameter.
 6. To verify that your SQL is still valid, click **Validate SQL**.
 7. If the SQL is valid, click **Create Dataset**.

Parameterize dataset with a timestamp

Timestamp parameters can be helpful when you want to filter datasets based on date and time format, time zone, or exact and relative start time. You can apply timestamp parameters based on the specific region or time zone for which the data is generated.

Steps:

1. Create a custom dataset using SQL. For more information, see [Create a Custom SQL Dataset](#) above.
2. In the Create Dataset with SQL window, enter a `SELECT*` statement to retrieve data from the specified table. Click **Validate SQL** to verify that the query is properly specified.
3. Now, highlight the part of the query that you wish to parameterize. Click the Timestamp icon.

The screenshot shows the 'Create Dataset with SQL' dialog box. The connection is set to 'postgres'. The SQL query is 'SELECT col1, col2 from'. A 'Timestamp Parameter' dialog is open over the query. It has fields for 'Timestamp format' (set to 'YYYY-MM-DD_hh_mm'), 'Timestamp value' (set to 'Relative to job start date'), and a dropdown for 'minus' (set to 'minus') and 'days' (set to '1'). Below these fields, it says 'E.g., for a job start time 2020-11-30 12:48, the recorded timestamp would be 2020-11-29_12_48.' and 'America/Los_Angeles time zone. [Change](#)'. At the bottom of the dialog are 'Cancel' and 'Save' buttons. The main dialog has 'Validate SQL', 'Cancel', and 'Create Dataset' buttons at the bottom.

Figure: Define Timestamp Parameter

4. In the Timestamp Parameter dialog, enter the following details:
 - i. **Timestamp format:** Specify the format for timestamp values.
 1. Example: `YYYY-MM-DD_hh_mm`.
 2. Datetime values can be expressed as either date or time elements.
 - ii. **Timestamp value:** Select the value to record in the path:
 1. **Exact job start date:** recorded timestamp in path is the start time of the job.
 2. **Relative to the job start date:** recorded timestamp in path is relative to the start time of the job according to the settings that you specify here.
 - iii. **Time zone:** Click **Change** to change the time zone recorded in the timestamp.
 1. Example: `America/Los Angeles` or `Asia/Calcutta`.
 2. For more information on the available time zones, see [Supported Time Zone Values](#).
5. Click **Save** to save the parameter.
6. To verify that your SQL is still valid, click **Validate SQL**.
7. If the SQL is valid, click **Create Dataset**.

Examples

In the following examples, you can see how dataset parameters can be used to pre-filter rows or parameterize the tables to include in your dataset.

NOTE: The syntax in these examples uses PostgreSQL syntax. The syntax that you use must match the requirements of the target database system.

Pre-filter rows from a table

Suppose you have a set of orders in a single table: `myOrders`. From this table, you want to be able to import a dataset that is pre-filtered for values in the customer identifier (`custId`) column. The following might be a query that you use for the `customerOrders` dataset to retrieve the orders for `custId=0001` from the `myOrders` table in the `transactions` database:

```
SELECT "custId","ordDate","prodId","ordQty","unitPrice" FROM "transactions"."myOrders" WHERE "custId" = "0001"
```

Tip: This example uses a variable parameter.

Steps:

In this case, you can do the following:

1. In the Create Dataset with SQL window, specify the above query. Click **Validate SQL** to verify that it works.
2. Now, highlight the value `0001`.
3. Select the Variable icon.
4. Specify your variable:
 - a. **Name:** `myCustId`
 - b. **Default Value:** `0001`
5. Click **Save**.
6. Before you create the dataset, validate the SQL.

Using the parameter:

- When the job is executed, the `customerOrders` dataset is pre-filtered to retrieve the data for `custId=0001` by default.
- You can override this variable value as needed:
 - At the flow level, you can define an override for the `myCustId` variable. For example, you can set the variable value to: `0002`. Whenever a job is run on this imported dataset, the value `0002` is passed to the running environment, which retrieves only the rows in the table where `custId=0002`.
 - When you run the job through the application, you can specify an override to the variable. This override takes precedence over the flow that was set at the flow level. So, for a specific run, you can set the value to `0003`, generating results for `custId=0003` only.
 - Then, the next time that a job is run from the flow using the dataset, the flow override value (`0002`) is used.

Run a weekly job on daily tables

Suppose you have a database that captures log data into separate tables for each date. Each table is named according to the following pattern:

```
20201101-ServerLogs
20201102-ServerLogs
20201103-ServerLogs
```



```
20201104-ServerLogs
20201105-ServerLogs
```

Once per week, you want to run a job to ingest and process the log entries from the preceding week.

The following could be a query that you use to retrieve all columns from a single file:

```
SELECT * FROM "logs"."20201101-ServerLogs"
```

Tip: This example uses a timestamp parameter.

Steps:

In this case, you can do the following:

1. In the Create Dataset with SQL window, specify the above query. Click **Validate SQL** to verify that it works.
2. Now, highlight the value `20201101`.
3. Select the Timestamp icon.
4. Specify your variable:
 - a. **Timestamp Format:** `YYYYMMDD`
 - b. **Timestamp Value:** `Relative to job start date`
 - i. Select `minus, 7, days`.
5. Click **Save**.
6. Before you create the dataset, validate the SQL.

Using the parameter:

When the job is executed, the imported dataset includes all of the tables whose timestamp format is within 7 days of the time when the job was started.

- You may need to modify the time zone setting on the Timestamp parameter if the log files were recorded using a different time zone.
- Typically, jobs created on a dataset like this one are executed according to a schedule.
 - For scheduled jobs, the value that is used for the job start date is the timestamp for when the job was scheduled to execute. It's possible that delays in starting the job could create a difference in the timestamps.
 - For more information, see *Schedule a Job*.

Parameterize entire query

You can turn the entire query of your custom SQL statement into a parameter. When you create your dataset with SQL, instead of entering any SQL in the window, create a variable parameter. For example, your parameter could be like the following:

- **Name:** `selectCustomersTable`
- **Value:**

```
SELECT * from "MDM"."customers"
```

If the SQL validates, then you can create the imported dataset using only this parameter.

How to use this parameter:

- By default, when the dataset is imported, all of the columns from the `customers` table are imported.
- As needed, you can configure overrides at the flow- or job-level to, for example, import only select columns. In the override, you specify the list of columns to gather only the data required for your needs.

Tip: This example uses a variable parameter.

Export Macro

As needed, you can export a macro from Designer Cloud Powered by Trifacta® Enterprise Edition. An exported macro is stored in a JSON file that contains all of the information required to use the macro in any instance of the product.

NOTE: Only the creator of a macro can export it.

Exported macros can be imported into the same system or different systems. Macro export is useful for:

- Backups of work in progress

You cannot import macros into an earlier release of the product.

- Archiving of completed development work
- Migrating macros from one instance to another

Export

Steps:

1. From the left navigation bar, select **Library**.
2. In the Library page, click **Macros**.
3. In the Macro page, locate the macro that you wish to export. In its context menu, select **Export**.
4. The JSON file is downloaded to the default download location on your local desktop.

When you import a macro, you import this JSON file.

Export via API

You can export macro definitions using the APIs.

Tip: This method is useful for publishing macro definitions across all deployments in your organization.

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/getMacroPackage>

Import Macro

Contents:

- *Limitations*
 - *Import*
 - *Import via API*
-

A macro that has been exported from the Designer Cloud Powered by Trifacta® Enterprise Edition can be imported back into the product.

A **macro** is a reusable set of steps that are specified from within a recipe. For more information, see *Overview of Macros*.

Limitations

You cannot import macros that were exported from a later release of the product.

- You cannot modify the macro definition JSON file outside of Designer Cloud Powered by Trifacta Enterprise Edition.
- If you are importing a flow into a Production instance of the platform, any macros referenced in the imported flow are expanded into their original steps in the recipes where they are referenced.
 - Macros cannot be imported, referenced, or viewed directly in a Production instance.
 - A Production instance is available only if you have enabled the Deployment Manager. For more information, see *Overview of Deployment Manager*.

Import

Tip: If you re-import a macro into the same instance that still contains the source macro, the imported version is named the same as the source and automatically versioned for you.

Steps:

1. Export the macro from the source system. See *Export Macro*.
2. Login to the import system, if needed.
3. In the left nav bar, click **Library**.
4. In the Library page, click **Macros**.
5. In the Macros page, click **Import Macro**.
6. Select the JSON file containing the exported macro.

Tip: You can import multiple macros at the same time. Select each JSON file in the dialog box that you wish to import. Press **CTRL/CMD** + click or **SHIFT** + click to select multiple macros for import.

7. Click **Open**.

The macro is imported and available in the Macros page.

To use an imported macro, enter `macro` in the Search panel in the Transform Builder. Select your macro and modify any macro inputs. For more information, see *Apply a Macro*.

Import via API

If you have exported your macro using the APIs, you can import it into a new environment. For more information, see

<https://api.trifacta.com/ee/es.t/index.html#operation/importMacroPackage>

Create Dataset with SQL

Contents:

- *Limitations*
 - *General*
 - *Single Statement*
 - *Multi-Statement*
 - *Enable*
 - *Use*
 - *Create with Variables*
 - *Create with timestamp parameter*
 - *SQL Validation*
 - *SQL Syntax*
 - *Troubleshooting*
 - *Snowflake*
-

As needed, you can insert custom SQL statements as part of the data import process. These custom SQL statements allow you to pre-filter the rows and columns of relational source data within the database, where performance is faster. This query method can also be used for wider operations on relational sources from within the Designer Cloud powered by Trifacta® platform .

Limitations

General

All queries are blindly executed. It is your responsibility to ensure that they are appropriate. Queries like `DELETE` and `DROP` can destroy data in the database. Please use caution.

NOTE: Column names in custom SQL statements are case-sensitive. Case mismatches between SQL statement and your datasource can cause jobs to fail.

- SQL statements are stored as part of the query instance for the object. If the same query is being made across multiple users using private connections, the SQL must be shared and entered by individual users.

NOTE: If a dataset created from custom SQL is shared, collaborators are not permitted to edit the custom SQL.

- Each statement must be terminated with a semi-colon (`;`) and a newline:

```
SELECT * FROM myDB.myTable;
```

- SQL statements must be valid for the syntax of the target relational system.
- If you modify the custom SQL statement when reading from a source, all samples generated based on the previous SQL are invalidated.
- Declared variables are not supported.
- Common Table Expressions (CTEs) are not supported.
- For each SQL statement, all columns must have an explicit name. Example:

- Function references such as:

```
UPPER(col)
```

- Must be specified as:

```
UPPER(col) as col_name
```

- When using custom SQL to read from a Hive view, the results of a nested function are saved to a temporary name, unless explicitly aliased.
 - If aliases are not used, the temporary column names can cause jobs to fail, on Spark in particular.
 - For more information, see *Hive Connections*.

Single Statement

The following limitations apply to creating datasets from a single statement.

1. All single-statement SQL queries must begin with a `SELECT` statement.
2. Selecting columns with the same name, even with " * ", is not supported and generates an ambiguous column name error.

Tip: You should use fully qualified column names or proper aliasing. See *Column Aliasing* below.

3. Users are encouraged to provide fully qualified path to table being used. Example:

```
SELECT "id", "value" FROM "public"."my_table";
```

4. You should use proper escaping in SQL.

Multi-Statement

These limitations apply to creating datasets using a sequence of multiple SQL statements.

NOTE: Use of multiple SQL statements must be enabled. See *Enable Custom SQL Query*.

1. **Repeatable:** When using multi-statements, you must verify that the statements are repeatable without failure. These statements are run multiple times during validation, datasets creation, data preview, and opening the dataset in the Transformer page.

NOTE: To ensure repeatability, any creation or deletion of data in the database must occur before the final required `SELECT` statement.

2. **Line Termination:** Each statement must terminate with a semi-colon and a new line. Example:

```
SELECT * FROM transactions.orders;
SELECT custId,custName FROM master.customers;
```

3. **Validation:** All statements are run immediately when validating or creating dataset.

NOTE: No DROP or DELETE checking is done prior to statement execution. Statements are the responsibility of the user.

4. **SELECT requirement:** In a multi-statement execution, the last statement must be a SELECT statement.
5. **Database transactions:** All statements are run in a transaction. DDL statements in most dialects (vendors) can't be run within a transaction and might be automatically committed by the driver.

Enable

Steps:

1. You apply this change through the *Workspace Settings Page*. For more information, see *Platform Configuration Methods*.
2. Locate the following setting:

Enable custom SQL Query

Setting	Description
enabled	Set to <code>true</code> to enable the ability to create datasets using customized SQL statements. By default, this feature is enabled.

Use

To use, please complete the following steps.

Steps:

1. In the Library page, click **Import Data**.
2. In the Import Data page, select a connection.
3. Within your source, locate the table from which you wish to import. Do not select the table.
4. Click the Preview icon to review the columns in the dataset.

Tip: You may wish to copy the database, table name, and column names to a text editor to facilitate generating your SQL statement.

5. Click **Create Dataset with SQL**. Enter or paste your SQL statement.

Through the custom SQL interface, it is possible to enter SQL statements that can delete data, change table schemas, or otherwise corrupt the targeted database. Please use this feature with caution.

NOTE: If this button is disabled and you have enabled the custom SQL feature, the connection that you are using may lack credentials. Please review the connection definition.

Connection: hive

```
1 SELECT INST#, BUCKET# FROM "AUDSYS"."CLI_SWP$7395268a$1$1"
```

[Validate SQL](#)

☒ Infer column data types

[Cancel](#) [Create Dataset](#)

Figure: Create Dataset with SQL dialog

- a. To test the SQL, click **Validate SQL**. For details, see below.
- b. To apply the SQL to the import process, click **Create Dataset**.
6. The customized source is added to the right panel. To re-edit, click **Custom SQL**.
7. Complete the other steps to define your imported dataset.
8. When the data is imported, it is altered or filtered based on your SQL statement.

Create with Variables

If parameterization has been enabled, you can specify variables as part of your SQL statement. Suppose you had table names like the following:

```
publish_create_all_types_97912510
publish_create_all_types_97944183
publish_create_all_types_14202824
```

You can insert an inline variable as part of your custom SQL to capture all of these variations.

Connection: hive

```
1 SELECT * from `default`.`publish_create_all_types` <> iid`
```

<> Variable

Name

Default value

[Cancel](#) [Save](#)

[Validate SQL](#)

☒ Infer column data types

[Cancel](#) [Create Dataset](#)

Figure: Insert variables in your custom SQL

In the above, custom SQL has been added to match the first example table. When the value is highlighted and the icon is clicked, the highlighted value is specified as the default value.

Tip: Type `env.` to see the environment parameters that can be applied. These parameters are available for use by each user in the environment.

Provide a name for the variable, and click **Save**.

Through the Run Job page, you can specify overrides for the default value, so the same job definition can be used across all matching tables without much modification.

Create with timestamp parameter

You can insert a timestamp parameter into your custom SQL. These parameters are used to describe timestamp formats for matching timestamps relative to the start of the job at the time of execution.

NOTE: A SQL timestamp parameter only describes the formatting of a timestamp value. It cannot be used to describe actual values. For example, you cannot insert fixed values for the month to parameterize your input using this method. Instead, parameterize the input using multiple input variables, as described in the previous section.

NOTE: Values for seconds in a SQL timestamp parameter are not supported. The finest supported granularity is at the minutes level.

NOTE: When the dataset is created, the current date is used for comparison, instead of the job execution date.

In the following example, the timestamp parameter has been specified as `YYYY-MM-DD`:

```
SELECT * FROM <YYYY-MM-DD> ;
```

If the job executes on May 28th, 2019, then this parameter resolves as `2019-05-28` and gathers data from that table.

Create Dataset with SQL ✕

Connection: hive

1 `SELECT 1 as`

Timestamp Parameter

Timestamp format

Timestamp value

Exact job start date ▼

E.g., for a job start time 2019-10-18 02:39, the recorded timestamp would be 2019-10-18.

America/Los_Angeles time zone. [Change](#)

Cancel Save

Validate SQL

☒ Infer column data types

Cancel Create Dataset

Figure: Insert timestamp parameter

Steps:

1. Click the Clock icon in the custom SQL dialog.
2. **Timestamp format:** You can specify the format of the timestamp using supported characters.

Tip: The list and definition of available tokens is available in the help popover.

3. **Timestamp value:** Choose whether the timestamp parameter is to match the exact start time or a time relative to the start of the job.

Tip: You can use relative timestamp parameters to collect data from the preceding week, for example. This relative timestamp allows you to execute weekly jobs for the preceding week's data.

4. To indicate that the timestamps are from a timezone different from the system timezone, click **Change**.
5. To save the specified timestamp parameter, click **Save**.

SQL Validation

You cannot create a SQL-based dataset if any of your SQL statements do not pass validation. Errors must be corrected in the SQL or in the underlying database.

- All `SELECT` statements are planned, which includes syntactical validation. However, these statements are not executed. Validation should be a matter of a few seconds.
- For multi-line statements, all non-`SELECT` statements are planned and executed. The final `SELECT` statement is only planned.

NOTE: For multi-line SQL statements, validation may take longer to complete if the non-`SELECT` statements require significant time to execute.

SQL Syntax

For more information on SQL syntax and supported variations, see *Supported SQL Syntax*.

Troubleshooting

Snowflake

Selecting time zone data returns null values in profiling and fails in publishing

When you import a column from Snowflake that contains time zone information, you may see the following behavior:

- Sampled data appears to import correctly into the Transformer page for the TIMESTAMP-based column.
- When a job is run, the visual profile for the output column based on this data indicates null values.
- When the data is published back to Snowflake, the publishing job fails.

The above issue is caused by the following:

- When data is imported into the Transformer page, it is automatically converted to UTC timezone during the JDBC ingestion step for displaying the sample in the application. This ingestion process is called by the application and outside of the application's control.
 - During this ingestion process, some auto-recognition and conversion to UTC of Datetime values is applied to the sample for display.
 - Example: You design a recipe step to parse the following Datetime format: 2020-10-11 12:13:14., which has been auto-converted to UTC.
- When a job is run:
 - The application instructs Snowflake to unload the entire dataset from Snowflake and write it the target location, bypassing this automatic conversion process.
 - The recipe that was created to handle the data in the sample does not properly handle the data that is directly unloaded from Snowflake.
 - In the previous example: The Datetime parsing in your recipe may receive an input that looks very different from what you parsed in the displayed sample: 2020-10-11 14:13:14 CEST.

Solution:

For a time stamp with a time zone, you must wrap your reference to it like the following:

```
TO_TIMESTAMP(CONVERT_TIMEZONE('UTC', <timestamp_column_or_function>))
```

Suppose your query was the following:

```
SELECT *, CURRENT_TIMESTAMP() AS current_time FROM MY_TABLE;
```

To address this issue, the query needs to be rewritten as follows:

```
SELECT *, TO_TIMESTAMP(CONVERT_TIMEZONE('UTC', CURRENT_TIMESTAMP())) AS current_time FROM MY_TABLE;
```

When the above wrapper function is applied, the data is imported normally and validated and published as expected.

Connections

Contents:

- *Not Covered*
 - *Hadoop*
 - *AWS*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

This section covers how to configure connections between the Designer Cloud powered by Trifacta® platform and your sources of data. Depending on the connection type, data can be imported, exported, or both through a defined and valid connection.

Many of these connections can be created from the Designer Cloud application directly. In some cases, additional configuration is required outside of the application.

Not Covered

This guide does not cover connection types that are deeply tied to a specific infrastructure. The following connection types are described in the appropriate Configuration Guide.

Hadoop

- *Hive Connections*

AWS

- *S3 Access*
- *Amazon Redshift Connections*

Connection Basics

Contents:

- *Basic Import Process*
- *Import Data*
- *Default Connections*

In the Designer Cloud powered by Trifacta® platform , you can quickly connect to default datastores and import your file- or table-based datasets for use in the Designer Cloud application . As needed, you can create connections to other datastores to which you have access and import from them.

Basic Import Process

There are multiple ways to import data into the Designer Cloud powered by Trifacta platform , but the most direct way is to import through the Library, where all imported datasets are stored. In the Designer Cloud application , click the **Library** icon in the left nav bar.

Name	Owner	In flows	Source	Last updated
USDA_Farmers_Market_2014.avro	Steve Olson	1	TFS	Last Thursday at 3:48 PM
POS-schema - 2.csv	Steve Olson	1	TFS	Last Thursday at 11:42 AM
POS-r01.csv	Steve Olson	1	TFS	Last Thursday at 11:42 AM
POS-r02.csv	Steve Olson	1	TFS	Last Thursday at 11:41 AM
REF_CAL.csv	Steve Olson	1	TFS	Last Thursday at 11:41 AM
REF_PROD.csv	Steve Olson	1	TFS	Last Thursday at 11:41 AM
POS-r03.csv	Steve Olson	1	TFS	Last Thursday at 11:41 AM
expeditions- 1960s.csv	Steve Olson	1	TFS	08/06/2021
climbs2014.csv	Steve Olson	1	TFS	08/06/2021
rainier_weather.csv	Steve Olson	1	TFS	08/06/2021
2015_orders.csv	Steve Olson	1	TFS	07/26/2021
2014_orders.csv	Steve Olson	1	TFS	07/26/2021
Clothing_customers.csv	Steve Olson	1	TFS	07/26/2021

Figure: Library Page

Import Data

The Library displays all of the datasets to which you have access. To begin, click **Import Data**.

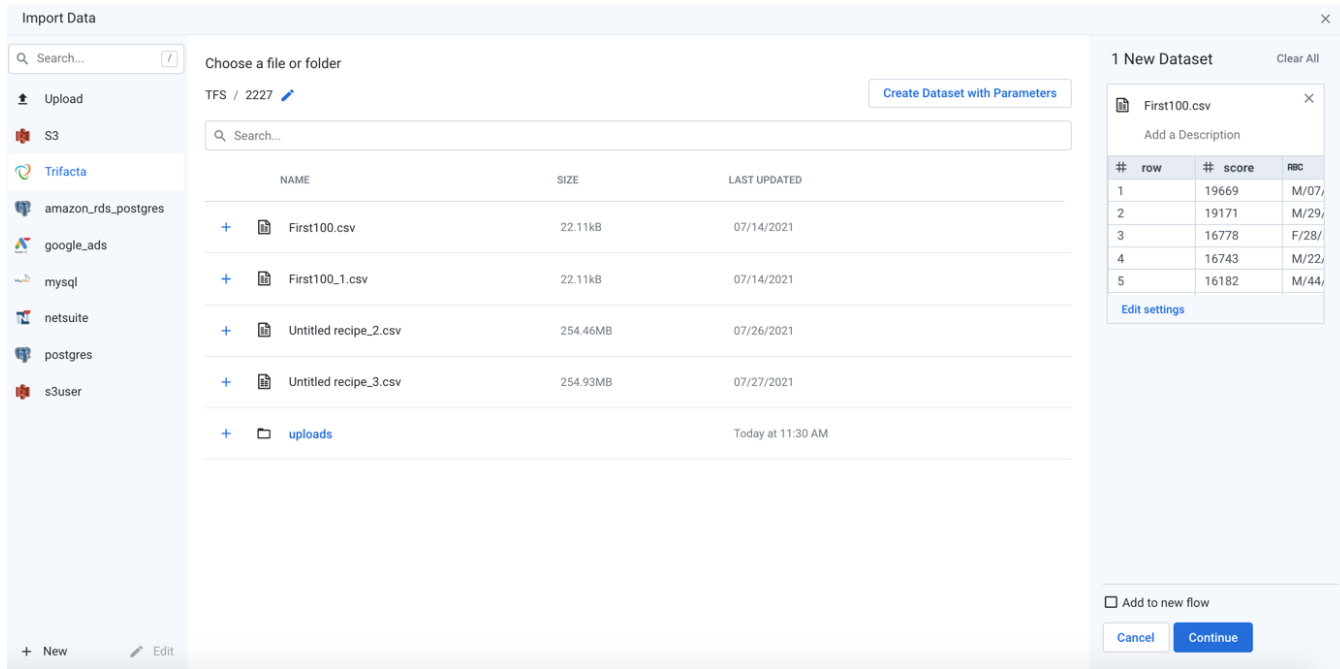





Figure: Import Data Page

When you import data, you are selecting a connection to use, which allows you to begin navigating the connected datastore. When you locate the file or table assets, you select it for import.

NOTE: When you import a dataset, you are creating a reference to that source dataset. Data is not actually imported until the Designer Cloud powered by Trifacta platform needs to use the data for job execution. Source data is never modified.

Default Connections


Designer Cloud Powered by Trifacta Enterprise Edition provides pre-configured connections to storage. Depending on your environment, you may be able to immediately access data through the following datastores, which are represented by icons in the Import Data page:

icon	Connection	Description
 Upload	Upload	Upload a file from your local desktop. Tip: You can always upload a file from your local desktop for immediate use in the application. Details are below.
 S3	File-based storage	The Designer Cloud powered by Trifacta platform provides a connection to a primary file-based storage system. Also known as the base storage layer, this storage can be used for hosting datasets for import. Uploaded files are stored on the base storage layer, as well as dataset samples.
 postgres	Table-based storage	Your environment may be provided access to a table-based datastore, such as BigQuery or Redshift. From these datastores, you can select the tables or views to which you have access for import as datasets in the Designer Cloud powered by Trifacta platform .
+ New	Other storage	You can create connections to other file- or table-based storage systems. See below for details.

Connections Page

Contents:

- *Top Bar*
- *Connection Context Menu*
- *Connection Details Panel*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the Connections page, you can add new connections or modify the connections that you have already created. From the left nav bar, click the Connections icon.

NOTE: Access to the Connections page in the application and privileges on connections is governed by roles in your workspace. For more information, please contact your workspace administrator.

Connections [Create Connection](#)

Search connections

All connections Owned by me Shared with me


Name	Type	Shared	Owner	Last updated	
 s3user	s3user	Private		04/15/2021	Share ...

Figure: Connections page

Fields:

- **Name:** Display name for the connection.

You can hover over the shared icon link next to the connection name to view the name of shared users (up to three shared user) and the total number of shared users. Also, when you click the shared icon link, the share dialog is displayed.

NOTE: If the connection has been shared, you can review whether its credentials have also been shared.

- **Type:** The type of connection.

NOTE: After you create a connection, you cannot modify its type.

For more information, see *Connection Types*.

- **Shared:** Review the sharing status of the connection:
- **Global** - connection has been shared with all users of the workspace.

NOTE: To make a global connection private, you must delete the connection and recreate it.

- **X Users:**
 - If this value is 1, the connection is private.
 - If this value is greater than 1, the connection has been shared. Click the link in this column to review sharing status. See *Share Connection Dialog*.

Top Bar

- **Create:** Click **Create Connection** to create a new connection. See *Create Connection Window*.
- **Filter:** In separate tabs, you can review connections that you own, that are shared with you, or all connections to which you have access.
- **Search:** Search connections by name.
- **Review details:** Select a connection or click the icon to review details through the right-side panel.

Connection Context Menu

- **Share:** For connections that you own, you can modify the sharing status of them. See *Share Connection Dialog*.
- **View Details:** Open the details of the connection in the side panel. See below.
- **Edit:** If you own the connection, you can review and modify the connection.
 - If the connection has been shared with you, you can edit it to modify the credentials.
 - Administrators can edit public connections.
 - See *Create Connection Window*.
- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user. See *Transfer Asset Ownership*.
- **Delete:** Delete the connection.


NOTE: This option is only available to the connection owner if the connection is not used for any datasets.

Connection Details Panel

When a connection is selected, you can review its details and make modifications as needed through the panel on the right.

Connection Details

×

 postgres

Edit Connection

...

Connection Type

postgres

Shared

Private

Owner

SteveO

Created

Today at 5:28 PM

Updated

Today at 5:28 PM

Updated by

SteveO

Server Information

Host

myHost

Port

5432

SSL

Disabled

Database

myDB

Username

myUserId

Password

.....

Figure: Connection Details panel

Key Fields:

- **Shared:** Number of users sharing the connection. If a link is present, click it to modify sharing of the connection. See *Share Connection Dialog*.
- **Server Information:** For server-based connections, you can review the connection properties.

Actions:

- **Edit Connection:** If you own the connection, you can review and modify the connection.
 - If the connection has been shared with you, connection properties are read-only.
 - See *Create Connection Window*.
- **Share:** You can share connections that you own or that are shared with you. See *Share Connection Dialog*.
- **Transfer ownership:** (Available to owner or admin only) Transfer ownership of this asset to another user. See *Transfer Asset Ownership*.
- **Delete:** Delete the connection.

Deleting a connection cannot be undone.

Create Connection Window



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the Create Connection window, you can create and edit connections between the Designer Cloud powered by Trifacta® platform and remote storage.

NOTE: Access to this page in the application and privileges on its related objects is governed by roles in your workspace. For more information, please contact your workspace administrator.

This window is available from the following locations:

- **From Import Data page:** By default, the window displays connections that support import. Deselect the checkbox to display all available connection types.
- **From Run Job page:** When you add a new connection as part of a publishing action, the window displays connections that support publishing by default.
- **From the Connections page:** All available connections are displayed.

NOTE: Some connection types may not be available for your environment.

NOTE: In your environment, creation of connections may be limited to administrators only. For more information, please contact your Trifacta administrator.

Tip: Administrators can edit any public connection.

General Connection Notes:

- After you create a connection, you cannot change its connection type. You must delete the connection and start again.
- Connections can be created, managed, shared, and deleted through the Connections page. See *Connections Page*.

Database Connection Notes:

- Database connections cannot be deleted if their databases host imported datasets that are in use by the Designer Cloud powered by Trifacta platform. Remove these imported datasets before deleting the connection.
- Jobs created for datasets sourced from a database cannot be executed on a Spark-based running environment.

Connection Type

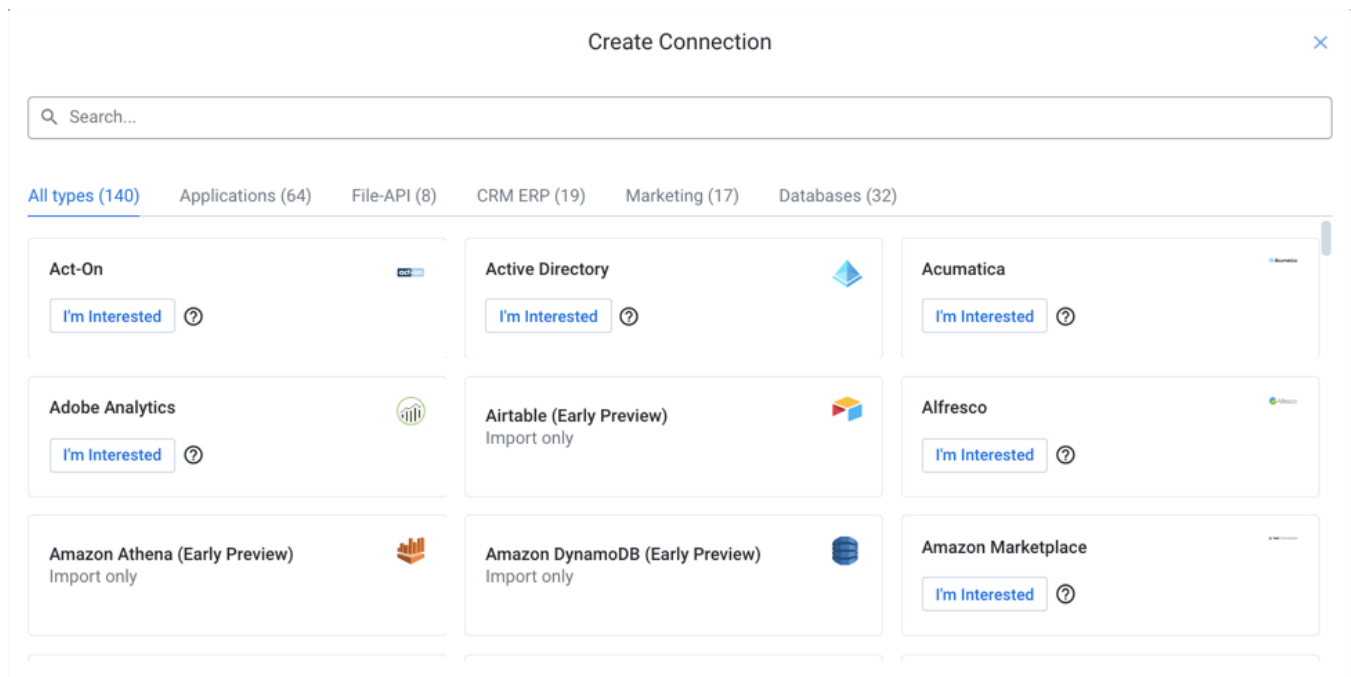


Figure: Connection Type window

In the Connection Type window, you search or browse for the type of connection to create.

Actions:

- Use the Search bar to perform real-time searches of connection types.
- Click one of the categories to browse for connection types that apply to the listed environment.
- Select the type of connection to continue:
 - **Import only** - Connection can be used only to import data into the platform.
 - **Publish only** - Connection can be used only to publish data from the platform to the connection target.
 - **Import and publish** - Connection can be used to import data and to publish your outputs.

For more information on these connections, See *Connection Types*.

Create Connection

Create Connection

PostgreSQL

Import and publish

SERVER INFORMATION

Host

myserver

Port

5432

Connect String Options (optional)

☒ Enable SSL

Database

myDB

User Name

dbuser

Password

Test Connection

< Back

Cancel

Create

Figure: Create Connection Window

Property	Description
Host	Host of the database.
Port	Port by which to access the database host. Default values are pre-populated based on the connection type you selected.
Connect String Options	(optional) If access to the database requires special connection string options, you may paste or enter them here. You only need to provide the parameter and string value. Example: <div>" ;transportMode=http;httpPath=cliservice"</div>
Enable SSL	To connect using SSL, click this checkbox. If this checkbox is not present, SSL connections for this database type are not supported or are required:

	<ul style="list-style-type: none"> • SSL connections are not supported for SQL Server or Hive. • SSL connections are required for Redshift and SQL DW. <p>No additional Connect String Options are required for supported database vendors.</p> <div> NOTE: The database must be configured to receive SSL connections. </div>
Service Name	(Oracle only) Name of the service. For example, enter <code>orcl</code> here.
Database	(PostgreSQL only) Name of the database to connect. The name of the default database is the username, so you should change this value in most cases.
Credential Type	<p>Depending on the type of datastore to which you are connecting, you may have multiple methods of providing credentials for authentication:</p> <ul style="list-style-type: none"> • <code>Basic</code> - Username and password credentials are provided as part of the connection definition. • <code>OAuth 2.0</code> - Connection accesses the datastore using OAuth 2.0 authentication. <div> NOTE: OAuth 2.0 authentication requires additional configuration. For more information, see <i>Enable OAuth 2.0 Authentication</i>. </div> <div> NOTE: For each type of connection that uses OAuth 2.0, you must create a client app and a client in the Designer Cloud application . See <i>Create OAuth2 Client</i>. </div> <div> NOTE: When you create the connection in this window for an OAuth 2.0 connection, you must click Authenticate, which uses the OAuth 2.0 client to connect to the app. This step is required. </div>
User Name	(basic credential type) Username to access the database. This value is encrypted for security.
Password	(basic credential type) Password for the specified user. This value is encrypted for security.
OAuth 2.0 Client	<p>(OAuth 2.0 credential type) Select the OAuth 2.0 client to use to connect to the datastore.</p> <div> NOTE: You must create a separate connection for each OAuth 2.0 client that is available in the drop-down list. </div>
Test Connection	When the above properties are specified, click Test Connection to validate that the Designer Cloud powered by Trifacta platform can connect to the database.If the connection test fails, your administrator may need to install a keyfile. See <i>Relational Access</i> .
Advanced Options: Default Column Data Type Inference	You can choose to enable or disable type inferencing for individual connections, when the connection is created or edited.The default setting for this parameter is defined at the global level. For more information, see <i>Configure Type Inference</i> .
Advanced Options: Enable SSH Tunneling	This feature is not available.
Connection Name	<p>Display name of the connection, which appears in the application.</p> <div> NOTE: This value must be unique among all connections. </div>
Connection Description	User-friendly description for the connection, which appears in the application.

When you've finished, click **Ok** to save the connection.


After you have created your connection, run a simple job on data sourced from it.

NOTE: You can make the connection available to all users by sharing it. See *Connections Page*.

Share Connection Dialog

Contents:

- *Actions*
- *Find Users*
- *Set Access Level*
- *Privileges*
- *Privacy*
- *Credentials*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the Share Connection dialog, users of the selected connection with appropriate privileges can modify who has access to the connection.

Tip: A workspace administrator has owner-level access to all connections in the workspace. However, a workspace admin cannot access or use a connection's credentials if those credentials were not shared by the owner of the connection. For more information, see *Workspace Admin Permissions*.

Share

×

Invite users

Editor

Share

Shared with (2 users)

A

Administrator (you)

A

Abarnan

Editor

Can read, update, and share

Viewer

Can read and share

Owner

Editor

🔒

Invite only

▼

Only users who have been invited can access this connection

🔑

Do not share credentials

▼

Users will need to provide their own credentials when accessing this connection

Figure: Share Connection Dialog

Actions

Find Users

Start typing names or email addresses of users to see matches.

NOTE: You may not be permitted to share objects with users who have not yet logged into the product.

Tip: You can paste a comma-separated list of email addresses to share to multiple users at the same time.

You may be able to browse a list of all usernames.

NOTE: This feature may need to be enabled in your environment. For more information, see *Workspace Settings Page*.

Set Access Level

As needed, you can configure the level of access to the connection for users with whom the connection is shared.

NOTE: You cannot set a user's access to a level that is higher than the limit set for the user at the workspace level. For example, if the user has Viewer access to connections at the workspace level, you cannot make the user an Editor on your connection.

NOTE: Administrators have owner-level access to all connections in the workspace or project. You do not need to share connections with them.

Privileges

- **Editor:**
 - User can use the connection to read data, update the connection, and share the connection.
 - User has all Viewer privileges.

NOTE: Editors cannot delete connections. Only the owner or an admin can delete a connection.

- **Viewer:**
 - User can use the connection to read data.
 - User can share connection.

For more information, see *Overview of Sharing*.

Privacy

- **Invite only:** Connection can be made available to other users only by invitation through this window.
- **Public :** Connection is available to all workspace users who can access connections.

NOTE: Only an administrator can make a connection public.

NOTE: After a connection is made public, it cannot be made private again. It must be deleted and recreated.

Credentials

By default, a connection is shared with credentials. Optionally, sharing of credentials can be disabled when sharing.

NOTE: The choice to share credentials or not is applied to all users with whom the connection has been shared, including users with whom the connection has been shared previously.

NOTE: Connections that use OAuth 2.0 authentication cannot be shared with credentials.

- **Share credentials:** When selected, the credentials that are specified in the owner's connection definition are shared with other users.

NOTE: Password values are always masked in the interface.

- **Do not share credentials:** When this option is selected, users of the shared connection must provide their own credentials.

NOTE: To use datasets previously imported through the shared connection, these credentials must provide access the source data. If shared credentials are removed from a connection, then any datasets imported through the connection are not accessible until credentials are provided. This may also apply to re-publishing previously generated results.

When credentials are shared:

NOTE: Users to whom credentials are shared cannot see any passwords in the Designer Cloud application .

- The credentials specified in the connection are shared to the users who are specified in the Share dialog for connections.
 - Users of a shared connection with credentials cannot insert their own credentials. They must create a new connection.
- Sharing of credentials may not guarantee access to the same locations as available to the owner.
Examples:
 - If your deployment uses Single Sign On, your enterprise login may provide access controls to the same resource that are different from the connection owner.
 - Network infrastructure may whitelist IP addresses for some users and block the same addresses for others.

- Depending on the datastore, folder or directory permissions may limit access.
- For more information, please contact your IT administrator.
- The owner of the connection can specify whether credentials are shared or not.
 - A workspace administrator has owner-level access to all connections in the workspace. However, a workspace admin cannot access or use a connection's credentials if those credentials were not shared by the owner of the connection.
- Shared users of the connection can share the connection if they have Editor privileges.

When credentials are not shared:

- Each user must provide credentials to use the connection.
 - A user's individual credentials may not provide read access to datasources, which may mean that imported datasets appear to be broken.
 - Individual credentials may not provide write access to the same output locations, which may cause jobs to fail.
- When sharing of credentials is disabled, shared users who share with other users cannot include credentials as part of the share.

Using Connections

This section contains some basic information on using common connections to various types of supported storage.

Using Databases

Contents:

- *Before You Begin*
 - *Access*
- *Storing Data in Relational Databases*
- *Reading from Database Tables and Views*
 - *Additional Notes on Database Views*
- *Running Jobs from Database Sources*
- *Writing to Databases*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

This section describes how you interact with your databases through the Designer Cloud powered by Trifacta® platform .

- Specific versions of each database are supported.
- Connections must be enabled and configured for each type of supported database.
- See *Connection Types*.

Before You Begin

- **Read Access:** Your database administrator must configure read permissions to the appropriate databases, tables and views for your use.

NOTE: To ensure that all user credentials used to access the database system are securely stored, you must first deploy the encryption key file to the Trifacta node. See *Relational Access*.

- **Write Access:** Some relational connection types support write access. For more information, see *Connection Types*.

Access

Database access is managed through connections.

- Individual users can create private connections through the application. See *Create Connection Window*.
- An administrator can make your connection public or create public connections through the application.

Storing Data in Relational Databases

NOTE: The **Designer Cloud powered by Trifacta platform** does not modify source data nor store transformed data in the relational systems. Datasets sourced from database tables or views are read without modification from their source locations.

Reading from Database Tables and Views

You can create a Trifacta dataset from a table or view stored in a connected database.

Tip: In some scenarios, you can improve performance of loading from database tables by creating a view on the table to restrict the amount of data loaded to only the required fields. Additional, you can pre-filter the dataset using custom SQL statements. See *Create Dataset with SQL*.

Additional Notes on Database Views

- Some metadata, such as row counts, is not available for database views.
- For complex view definitions that require significant processing on the database, there may be a significant delay when previewing the contents of those views. In some cases, the preview may time out waiting for the database to respond with the view contents.

For more information, see *Database Browser*.

Running Jobs from Database Sources

NOTE: When executing a job using a relational source, the job may fail if one or more columns has been dropped from the underlying source table.

Writing to Databases

Relational connections can be configured to support writing results back to the database.

NOTE: You can only write to databases from the Run Job page. You cannot ad-hoc publish to a relational database.

NOTE: When writing to a new table in a relational target, the first entry in any mapping is used for writing out the value. Subsequent entries in the mapping are used for validation only on writing to new tables.

Natively supported connection types are automatically enabled for writeback.

Using HDFS

Contents:

- *Uses of HDFS*
 - *Before You Begin Using HDFS*
 - *Secure Access*
 - *Storing Data in HDFS*
 - *Ingest Caching*
 - *Reading from Sources in HDFS*
 - *Creating Datasets*
 - *Writing Job Results*
 - *Creating a new dataset from results*
 - *Purging Files*
-

This section describes how you interact through the Designer Cloud powered by Trifacta® platform with your HDFS environment.

- HDFS is a scalable file storage system for use across all of the nodes (servers) of a Hadoop cluster. Many interactions with HDFS are similar with desktop interactions with files and folders. However, what looks like a "file" or "folder" in HDFS may be spread across multiple nodes in the cluster. For more information, see https://en.wikipedia.org/wiki/Apache_Hadoop#HDFS.

Uses of HDFS

The Designer Cloud powered by Trifacta platform can use HDFS for the following reading and writing tasks:

1. **Creating Datasets from HDFS Files:** You can read in from a data source stored in HDFS. A source may be a single HDFS file or a folder of identically structured files. See *Reading from Sources in HDFS* below.
2. **Reading Datasets:** When creating a dataset, you can pull your data from another dataset defined in HDFS. See *Creating Datasets* below.
3. **Writing Job Results:** After a job has been executed, you can write the results back to HDFS. See *Writing Job Results* below.

In the Designer Cloud application, HDFS is accessed through the HDFS browser. See *HDFS Browser*.

NOTE: When the Designer Cloud powered by Trifacta platform executes a job on a dataset, the source data is untouched. Results are written to a new location, so that no data is disturbed by the process.

Before You Begin Using HDFS

- **Read/Write Access:** Your Hadoop administrator must configure read/write permissions to locations in HDFS. Please see the HDFS documentation provided with your Hadoop distribution.

Avoid using `/trifacta/uploads` for reading and writing data. This directory is used by the Designer Cloud application.

NOTE: Use of HDFS in safe mode is not supported.

- Your Hadoop administrator should provide a place or mechanism for raw data to be uploaded to your Hadoop datastore.

- Your Hadoop administrator should provide a writeable home output directory for you, which you can review. See *Storage Config Page*.

Secure Access

Depending on the security features you've enabled, the technical methods by which Trifacta users access HDFS may vary. For more information, see *Configure Hadoop Authentication*.

Storing Data in HDFS

Your Hadoop administrator should provide raw data or locations and access for storing raw data within HDFS. All Trifacta users should have a clear understanding of the folder structure within HDFS where each individual can read from and write their job results.

- Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: The Designer Cloud powered by Trifacta platform does not modify source data in HDFS. Sources stored in HDFS are read without modification from their source locations, and sources that are uploaded to the platform are stored in `/trifacta/uploads`.

Ingest Caching

If JDBC ingest caching has been enabled, users may see a `dataSourceCache` folder in their browser. This folder is used to store per-user caches of JDBC-based data that has been ingested into the platform from its source.

NOTE: The `datasourceCache` folder should not be used for reading and writing of datasets, metadata, or results.

For more information, see *Configure JDBC Ingestion*.

Reading from Sources in HDFS

You can create a dataset from one or more files stored in HDFS.

NOTE: To be able to import datasets from the base storage layer, your user account must include the `dataAdmin` role.

Wildcards:

You can parameterize your input paths to import source files as part of the same imported dataset. For more information, see *Overview of Parameterization*.

Folder selection:

When you select a folder in HDFS to create your dataset, you select all files in the folder to be included. Notes:

- This option selects all files in all sub-folders. If your sub-folders contain separate datasets, you should be more specific in your folder selection.
- All files used in a single dataset must be of the same format and have the same structure. For example, you cannot mix and match CSV and JSON files if you are reading from a single directory.
- When a folder is selected from HDFS, the following file types are ignored:
 - `*_SUCCESS` and `*_FAILED` files, which may be present if the folder has been populated by Hadoop.

- If you have stored files in HDFS that begin with an underscore (`_`), these files cannot be read during batch transformation and are ignored. Please rename these files through HDFS so that they do not begin with an underscore.

Creating Datasets

When creating a dataset, you can choose to read data in from a source stored from HDFS or from a local file.

- HDFS sources are not moved or changed.
- Local file sources are uploaded to `/trifacta/uploads` where they remain and are not changed.

Data may be individual files or all of the files in a folder. For more information, see *Reading from Sources in HDFS*.

- In the Import Data page, click the HDFS tab. See *Import Data Page*.

Writing Job Results

When your job results are generated, they can be stored back in HDFS for you at the location defined for your user account.

- The HDFS location is available through the Output Destinations tab of the Job Details page. See *Job Details Page*.
- Each set of job results must be stored in a separate folder within your HDFS output home directory.
- For more information on your output home directory, see *Storage Config Page*.

If your deployment is using HDFS, do not use the `trifacta/uploads` directory. This directory is used for storing uploads and metadata, which may be used by multiple users. Manipulating files outside of the Designer Cloud application can destroy other users' data. Please use the tools provided through the interface for managing uploads from HDFS.

NOTE: Users can specify a default output home directory and, during job execution, an output directory for the current job. In an encrypted HDFS environment, these two locations must be in the same encryption zone. Otherwise, writing the job results fails with a `Publish Job Failed` error.

Access to results:

Depending on how the platform is integrated with HDFS, other users may or may not be able to access your job results.

- If user impersonation is enabled, results are written to HDFS through the HDFS account configured for your use. Depending on the permissions of your HDFS account, you may be the only person who can access these results.
- If user impersonation is not enabled, then each Trifacta user writes results to HDFS using a shared account. Depending on the permissions of that account, your results may be visible to all platform users.

Creating a new dataset from results

As part of writing job results, you can choose to create a new dataset, so that you can chain together data wrangling tasks.

NOTE: When you create a new dataset as part of your job results, the file or files are written to the designated output location for your user account. Depending on how your Hadoop permissions are configured, this location may not be accessible to other users.

Purging Files

Other than temporary files, the Designer Cloud powered by Trifacta platform does not remove any files that were generated or used by the platform, including:

- Uploaded datasets
- Generated samples
- Generated results

If you are concerned about data accumulation, please contact your HDFS administrator.

Using S3

Contents:

- *Uses of S3*
 - *Before You Begin Using S3*
 - *Secure Access*
 - *Storing Data in S3*
 - *Reading from Sources in S3*
 - *Creating Datasets*
 - *Writing Results*
 - *Creating a new dataset from results*
 - *Purging Files*
-

This section describes how you interact through the Designer Cloud powered by Trifacta® platform with your S3 environment.

- Simple Storage Service (S3) is an online data storage service provided by Amazon, which provides low-latency access through web services. For more information, see <https://aws.amazon.com/s3/>.

Uses of S3

The Designer Cloud powered by Trifacta platform can use S3 for the following tasks:

1. **Enabled S3 Integration:** The Designer Cloud powered by Trifacta platform has been configured to integrate with your S3 instance. For more information, see *S3 Access*.
2. **Creating Datasets from S3 Files:** You can read in source data stored in S3. An imported dataset may be a single S3 file or a folder of identically structured files. See *Reading from Sources in S3* below.
3. **Reading Datasets:** When creating a dataset, you can pull your data from a source in S3. See *Creating Datasets* below.
4. **Writing Results:** After a job has been executed, you can write the results back to S3.

In the Designer Cloud application, S3 is accessed through the S3 browser. See *S3 Browser*.

NOTE: When the Designer Cloud powered by Trifacta platform executes a job on a dataset, the source data is untouched. Results are written to a new location, so that no data is disturbed by the process.

Before You Begin Using S3

- **Access:** If you are using system-wide permissions, your administrator must configure access parameters for S3 locations. If you are using per-user permissions, this requirement does not apply. See *S3 Access*.

Avoid using `/trifacta/uploads` for reading and writing data. This directory is used by the Designer Cloud application.

- Your administrator should provide a writeable home output directory for you. This directory location is available through your user profile. See *Storage Config Page*.

Secure Access

Your administrator can grant access on a per-user basis or for the entire Designer Cloud powered by Trifacta platform.

The Designer Cloud powered by Trifacta platform utilizes an S3 key and secret to access your S3 instance. These keys must enable read/write access to the appropriate directories in the S3 instance.

NOTE: If you disable or revoke your S3 access key, you must update the S3 keys for each user or for the entire system.

Storing Data in S3

Your administrator should provide raw data or locations and access for storing raw data within S3. All Trifacta users should have a clear understanding of the folder structure within S3 where each individual can read from and write results.

- Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.
- The Designer Cloud application stores the results of each job in a separate folder in S3.

NOTE: The Designer Cloud powered by Trifacta platform does not modify source data in S3. Source data stored in S3 is read without modification from source locations, and source data uploaded to the Designer Cloud powered by Trifacta platform is stored in `/trifacta/uploads`.

Reading from Sources in S3

You can create an imported dataset from one or more files stored in S3.

NOTE: To be able to import datasets from the base storage layer, your user account must include the `dataAdmin` role.

NOTE: Import of glaciated objects is not supported.

Wildcards:

You can parameterize your input paths to import source files as part of the same imported dataset. For more information, see *Overview of Parameterization*.

Folder selection:

When you select a folder in S3 to create your dataset, you select all files in the folder to be included.

Notes:

- This option selects all files in all sub-folders and bundles them into a single dataset. If your sub-folders contain separate datasets, you should be more specific in your folder selection.
- All files used in a single imported dataset must be of the same format and have the same structure. For example, you cannot mix and match CSV and JSON files if you are reading from a single directory.

When a folder is selected from S3, the following file types are ignored:

- `*_SUCCESS` and `*_FAILED` files, which may be present if the folder has been populated by the running environment.

NOTE: If you have a folder and file with the same name in S3, search only retrieves the file. You can still navigate to locate the folder.

Creating Datasets


When creating a dataset, you can choose to read data in from a source stored from S3 or local file.

- S3 sources are not moved or changed.
- Local file sources are uploaded to `/trifacta/uploads` where they remain and are not changed.

Data may be individual files or all of the files in a folder. In the Import Data page, click the S3 tab. See *Import Data Page*.

Tip: Users can create secondary connections to specific S3 buckets. For more information, see *External S3 Connections*.

Full execution on Snowflake :

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

For S3 data sources that are written to Snowflake , you may be able to execute the job in Snowflake .

- You must enable the Full execution for S3 file option and configure the Snowflake flow optimizations. For more information, see *Flow Optimization Settings Dialog*.
- Additional configuration and limitations may apply. For more information, see *Snowflake Running Environment*.

Writing Results

When you run a job, you can specify the S3 bucket and file path where the generated results are written. By default, the output is generated in your default bucket and default output home directory.

- Each set of results must be stored in a separate folder within your S3 output home directory.
- For more information on your output home directory, see *Storage Config Page*.

NOTE: The `append` action is not supported when publishing to S3.

If Trifacta installation is using S3, do not use the `trifacta/uploads` directory. This directory is used for storing uploads and metadata, which may be used by multiple users. Manipulating files outside of the Designer Cloud application can destroy other users' data. Please use the tools provided through the Designer Cloud application interface for managing uploads from S3.

NOTE: When writing files to S3, you may encounter an issue where the UI indicates that the job failed, but the output file or files have been written to S3. This issue may be caused when S3 does not report the files back to the application before the S3 consistency timeout has expired. For more information on raising this timeout setting, see *S3 Access*.

Creating a new dataset from results

As part of writing results, you can choose to create a new dataset, so that you can chain together data wrangling tasks.

NOTE: When you create a new dataset as part of your results, the file or files are written to the designated output location for your user account. Depending on how your permissions are configured, this location may not be accessible to other users.

Purging Files

Other than temporary files, the Designer Cloud powered by Trifacta platform does not remove any files that were generated or used by the platform, including:

- Uploaded datasets
- Generated samples
- Generated results

If you are concerned about data accumulation, you should create a bucket policy to periodically backup or purge directories in use. For more information, please see the S3 documentation.

Using SQL DW

Contents:

- *Limitations*
 - *Uses of Azure Synapse Analytics (Formerly Microsoft SQL DW)*
 - *Before You Begin Using Azure Synapse Analytics (Formerly Microsoft SQL DW)*
 - *Secure Access*
 - *Storing Data*
 - *Reading Data*
 - *Writing to Azure Synapse Analytics (Formerly Microsoft SQL DW)*
-

This section describes how you interact through the Designer Cloud powered by Trifacta® platform with your Azure® Synapse Analytics (Formerly Microsoft® SQL DW)® data warehouse.

- Azure Synapse Analytics (Formerly Microsoft SQL DW) is a scalable data warehouse solution available through Microsoft Azure. For more information, see <https://docs.microsoft.com/en-us/azure/sql-data-warehouse/sql-data-warehouse-overview-what-is>.
- Azure Synapse Analytics (Formerly Microsoft SQL DW) connections can interact with data stored as managed tables or external tables.
- Azure Synapse Analytics (Formerly Microsoft SQL DW) connections can use dedicated or serverless SQL pools.
- For more information, see *Microsoft SQL Data Warehouse Connections*.

Limitations

- Azure Synapse Analytics (Formerly Microsoft SQL DW) connections are available only if you have deployed the Designer Cloud powered by Trifacta platform onto Azure.
- The defined length of a table row cannot exceed 1 MB.

NOTE: In this release, this connection cannot be created through the APIs. Please create connections of this type through the application.

Uses of Azure Synapse Analytics (Formerly Microsoft SQL DW)

The Designer Cloud powered by Trifacta platform can use Azure Synapse Analytics (Formerly Microsoft SQL DW) for the following tasks:

1. Create datasets by reading from Azure Synapse Analytics (Formerly Microsoft SQL DW) tables.
2. Write to Azure Synapse Analytics (Formerly Microsoft SQL DW) tables with your job results.
3. Ad-hoc publication of data to Azure Synapse Analytics (Formerly Microsoft SQL DW) .

Before You Begin Using Azure Synapse Analytics (Formerly Microsoft SQL DW)

- **Enable Access:** Integration requires the following:
 - Installation of the Designer Cloud powered by Trifacta platform on Microsoft Azure.
 - Either ADL or WASB is supported as the base storage layer. For more information, see *Set Base Storage Layer*.
- **Read Access:** Your administrator must configure read permissions. Your administrator should provide a database for upload.

- **Write Access:** You can write and publish jobs results to Azure Synapse Analytics (Formerly Microsoft SQL DW) .

Secure Access

These connections require SSL access.

Storing Data

Your Azure Synapse Analytics (Formerly Microsoft SQL DW) administrator should provide database access for storing datasets. Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: The Designer Cloud powered by Trifacta platform does not modify source data. Datasets sourced from Azure Synapse Analytics (Formerly Microsoft SQL DW) connections are read without modification from their source locations.

Reading Data

You can create a Trifacta dataset from a managed or external table through Azure Synapse Analytics (Formerly Microsoft SQL DW) .

For more information, see *Database Browser*.

Writing to Azure Synapse Analytics (Formerly Microsoft SQL DW)

You can write back data to Azure Synapse Analytics (Formerly Microsoft SQL DW) using one of the following methods:

NOTE: Writing and publishing to Azure Synapse Analytics (Formerly Microsoft SQL DW) is not supported if Azure AD SSO has been enabled.

- Job results can be written directly to Azure Synapse Analytics (Formerly Microsoft SQL DW) as part of the normal job execution. Create a new publishing action to write to Azure Synapse Analytics (Formerly Microsoft SQL DW) . See *Microsoft SQL Data Warehouse Table Settings*.
- As needed, you can publish results to Azure Synapse Analytics (Formerly Microsoft SQL DW) for previously executed jobs.
- For more information on how data is converted to Azure Synapse Analytics (Formerly Microsoft SQL DW) , see *SQL DW Data Type Conversions*.

Data Validation issues:

- No validation is performed for the connection and any required permissions during job execution. So, you can be permitted to launch your job even if you do not have sufficient connectivity or permissions to access the data. The corresponding publish job fails at runtime.
- No data validation is performed during writing and publication to Azure Synapse Analytics (Formerly Microsoft SQL DW) . Your job fails if the schema for the Trifacta dataset varies from the target schema.
- Prior to publication, no validation is performed on whether a target is a table or a view, so the job that was launched fails at runtime.

Connection Tasks

Create connections to your enterprise datastores and share them with other users of your workspace or project.

Configure Connectivity

Contents:

- *Enable*
 - *Data Service*
 - *Relational Features*
 - *Custom SQL Query*
 - *JDBC Ingestion*
 - *Append hadoop principal to logged queries*
 - *Enable Driver Logging*
 - *Configure Security*
 - *Enable SSO Connections*
 - *Type Inference*
 - *Enable OAuth 2.0 Connectivity*
-

This section covers the following areas around general connectivity of the Designer Cloud powered by Trifacta® platform .

Additional configuration may be required for individual connection types. For more information, see *Connection Types*.

Enable

The platform automatically enables connectivity to relational databases for reading in datasets and writing results back out.

NOTE: Relational connectivity requires the use of an encryption key file, which must be created and deployed before you create relational connections. For more information, see *Create Encryption Key File* in the Install Guide.

Data Service

The platform streams records from relational sources through the data service. These records are applied to transformation and sampling jobs on the Trifacta Photon running environment.

Tip: In general, you should not have to modify settings for the data service. However, if you are experiencing general performance issues or issues with specific connection types, you may experiment with settings in the data service.

For more information, see *Configure Data Service* in the Configuration Guide.

Relational Features

Custom SQL Query

To enhance performance of your relational datasets, you can enable the use of custom SQL queries against your relational datasources, which allows you to pre-filter your datasets before you ingest them into the platform. This feature is enabled by default, but additional configuration can be applied. See *Enable Custom SQL Query*.

JDBC Ingestion

By default, the platform ingests data from your relational datasources to the base storage layer for faster job execution. See *Configure JDBC Ingestion*.

Append hadoop principal to logged queries

Optionally, you can choose to enable appending the Hadoop principal as a comment to the SQL queries that are written to your database logs.

NOTE: This feature only applies if you are connecting to a Hadoop cluster. Otherwise, no value is inserted if this feature is enabled.

Example:

```
execute <unnamed>: SELECT * FROM "public"."artifacts" LIMIT 10 /* <hadoopPrincipal> */
```

In the above, the value of the Hadoop principal is written in the comment.

These types of queries are logged for the following basic activities:

- Data preview: when previewing data from a relational source, a query is executed against the database
- Data import: when selecting a table to import
- Data import using custom SQL:
 - Click Validate button.
 - Custom SQL execution.

This feature enables auditing of Trifacta user activities through your database logs in Hadoop-based environments.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following setting and set it to `true`:

```
"feature.addUserIdToSQLQuery.enabled": false,
```

3. After enabling the feature, locate the following setting, which defines the connection types for which the `userId` is written. Each connection type must be explicitly added to the list.
 - a. These are the default connection types to which it is applied:

```
"feature.addUserIdToSQLQuery.enabledConnectors": "teradata,sqlserver,oracle,db2,mysql",
```

- b. To apply this feature to other connection types, insert the connection type value to the comma-separated list. In the following example, PostgreSQL connections has been added to the list of default connection type:

```
"feature.addUserIdToSQLQuery.enabledConnectors": "teradata,sqlserver,oracle,db2,mysql,postgres",
```

- c. You can acquire the value to insert by reviewing a connection of the same type in the Designer Cloud application . When you select the connection in the Designer Cloud application , the appropriate value to insert is listed for **Connection type** in the connection information.

- d. If the above setting is set to an empty string (" "), the feature is applied to no connections and is effectively disabled.
4. Save your changes and restart the platform.

Enable Driver Logging

Optionally, you can enable the inclusion of log entries from the driver underlying a relational connection.

NOTE: This option applies only to relational connections that rely on CData drivers. Some connections may not support this option.

When you create or edit a relational connection, insert the following as part of the Connect String Options:

```
logfile=STDOUT://;verbosity=5;
```

Log entries are included in the `data-service.log` file is included in the standard Support Bundle. For more information, see *Support Bundle Contents*.

Configure Security

For more information, see *Configure Security for Relational Connections*.

Enable SSO Connections

If you have enabled Kerberos on the Hadoop cluster, you can leverage the Kerberos global keytab to enable SSO connections to relational sources. See *Enable SSO for Relational Connections*.

Type Inference

By default, the platform applies type inferencing to all imported datasources. However, for schematized sources, you may wish to disable type inferencing from the platform instead relying on the types provided from the source.

Tip: You can also toggle the use of type inferencing for individual connections or for individual imported datasets.

For more information, see *Configure Type Inference*.

Enable OAuth 2.0 Connectivity

Some supported relational datastores support authentication using OAuth 2.0.

- For each system to which you want to connect, you must create a client app in the target system. For more information, see *Enable OAuth 2.0 Authentication*.
- For a target system for which you have created a client app, you must create at least one client in the Designer Cloud application . For more information, see *Create OAuth2 Client*.

Relational Access

Contents:

- *Supported Relational Databases*
 - *Ports*
 - *Enable*
 - *Limitations*
 - *Execution at scale*
 - *Password Encryption Key File*
-

The Designer Cloud powered by Trifacta® platform can be configured to access data stored in relational database sources over JDBC protocol. When this connection method is used, individual database tables and views can be imported as datasets.

Supported Relational Databases

The Designer Cloud powered by Trifacta platform can natively connect to these relational database platforms. Natively supported versions are the following:

- Oracle 12.1.0.2
- SQL Server 12.0.4
- PostgreSQL 9.3.10
- Teradata 14.10+

NOTE: To enable Teradata connections, you must download and install Teradata drivers first. For more information, see *Enable Teradata Access*.

Additional relational connections can be enabled and configured for the platform. For more information, see *Connection Types*.

Ports

For any relational source to which you are connecting, the Trifacta node must be able to access it through the specified host and port value.

Please contact your database administrator for the host and port information.

Enable

This feature is enabled automatically.

To disable:

To prevent users from connecting to relational datasources for importing datasets and writing results, please complete the following configuration changes:

NOTE: Disabling this feature hides existing relational connections.

1. You apply this change through the *Workspace Settings Page*. For more information, see *Platform Configuration Methods*.

2. Locate the following setting:

```
Connectivity feature
```

3. Set this value to `Disabled`.

Disable relational publishing

By default, relational connections are read/write, which means that users can create connections that enable writing back to source databases.

- When this feature is enabled, writeback is enabled for all natively supported relational connection types. See *Connection Types*.
- Depending on the connection type, the Designer Cloud powered by Trifacta platform writes its data to different field types in the target database. For more information, see *Type Conversions*.
- Some limitations apply to relational writeback. See Limitations below.

As needed, you can disable this feature.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameter and set it to `false`:

```
"webapp.connectivity.relationalWriteback.enabled": true,
```

3. Save changes and restart the platform.

Publishing through relational connections is disabled.

Limitations

NOTE: Unless otherwise noted, authentication to a relational connection requires basic authentication (username/password) credentials.

- You cannot swap relational sources if they are from databases provided by different vendors. See *Flow View Page*.
- There are some differences in behavior between reading tables and views. See *Using Databases*.

Limitations on relational publishing:

When the relational publishing feature is enabled, it is automatically enabled for all platform-native connection types. You cannot disable relational publishing for Oracle, SQL Server, PostgreSQL, or Teradata connection types. Before you enable, please verify that all user accounts accessing databases of these types have appropriate permissions.

NOTE: Writing back to the database utilizes the same user credentials and therefore permissions as reading from it. Please verify that the users who are creating read/write relational connections have appropriate access.

- You cannot ad-hoc publish to a relational target. Relational publishing is only supported through the Run Job page.
- You write to multiple relational outputs from the same job only if they are from the same vendor.
 - For example, if you have two SQL Server connections A and B, you can write one set of results to A and another set of results to B for the same job.
 - If A and B are from different database vendors, you cannot write to them from the same job.

Execution at scale

Jobs for large-scale relational sources can be executed on the Spark running environment. After the data source has been imported and wrangled, no additional configuration is required to execute at scale.

NOTE: End-to-end performance is likely to be impacted by:

- streaming data volumes over 1 TB from the source,
- streaming from multiple concurrent sources,
- overall network bandwidth.

When the job is completed, any temporary files are automatically removed from HDFS.

For more information, see *Run Job Page*.

Password Encryption Key File

Relational database passwords are encrypted using key files:

- **Passwords in transit:** The platform uses a proprietary encryption key that is invoked each time a relational password is shared among platform services.
- **Passwords at rest:** For creating connections to your relational sources, you must create and reference your own encryption key file. This encryption key is accessing your relational connections from the web application. For more information, see *Create Encryption Key File*.

Enable Custom SQL Query

Contents:

- *Enable*
 - *Enable multi-statement*
 - *Configure query timeout*
 - *Use Custom SQL Queries*
-

To improve performance of your Hive or relational connections, custom SQL queries can be enabled to push the initial filtration of table rows and columns back the database, which is more efficient at performing this task. Instead of loading the entire table into the Designer Cloud® application and then performing the filtration through the Transformer page, you can insert basic SQL commands as part of your relational queries to collect only the rows and columns of interest from the source.

When enabled, custom SQL query is available for all relational sources.

Limitations

See *Create Dataset with SQL*.

Enable

Steps:

1. You apply this change through the *Workspace Settings Page*. For more information, see *Platform Configuration Methods*.
2. Locate the following setting:

```
Enable custom SQL Query
```

Setting	Description
enabled	Set to true to enable the SQL pushdown feature. By default, this feature is enabled.

Enable multi-statement

Optionally, you can enable the use of multiple statements in your SQL queries for imported datasets.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following setting:

```
"webapp.connectivity.customSQLQuery.enableMultiStatement": false,
```

Setting	Description
enableMultiStatement	<p>When set to <code>true</code>, you can insert multi-line statements in your SQL pushdown queries. The default is <code>false</code>.</p> <div> <p>NOTE: Use of multi-line SQL has limitations. See <i>Create Dataset with SQL</i>.</p> </div>

3. Save your changes and restart the platform.

Configure query timeout

As needed, you can configure the maximum permitted load time before timeout from the application. See *Configure Application Limits*.

Use Custom SQL Queries

When custom SQL query is enabled, you can enter customized SQL statements in the imported dataset page as part of the import process. See *Import Data Page*.

For examples, see *Create Dataset with SQL*.

After a dataset has been imported using custom SQL, you can edit the SQL as needed. See *Dataset Details Page*.

Configure JDBC Ingestion

Contents:

- *Recommended Table Size*
- *Performance*
- *Enable*
- *Configure*
 - *Configure Ingestion*
 - *Logging*
- *Monitoring Progress*
- *Logging*

This section describes some of the configuration options for the JDBC (relational) ingestion, which support faster execution of JDBC-based jobs.

Data ingestion works by streaming a JDBC source into a temporary storage space in the base storage layer to stage the data for job execution. The job can then be run on Photon or Spark. When the job is complete, the temporary data is removed from base storage or retained in the cache (if it is enabled).

- Data ingestion happens for Spark and Trifacta Photon jobs.
- Data ingestion applies only to JDBC sources that are not native to the running environment. For example, JDBC ingestion is not supported for Hive.
- Schema information is retained from the schematized source and is applied during publication of the generated results.
- Supported for HDFS and other large-scale backend datastores.

Data caching refers to the process of ingesting and storing data sources on the Trifacta node for a period of time for faster access if they are needed for additional platform operations.

Tip: Data ingestion and data caching can work together. For more information on data caching, see *Configure Data Source Caching*.

Job Type	JDBC Ingestion Enabled only	JDBC Ingestion and Caching Enabled
transformation job	Data is retrieved from the source and stored in a temporary backend location for use in sampling.	Data is retrieved from the source for the job and refreshes the cache where applicable.
sampling job	See previous.	<div>Cache is first checked for valid data objects. Outdated objects are retrieved from the data source.</div> <div>Retrieved data refreshes the cache.</div> <div>NOTE: Caching applies only to full scan sampling jobs. Quick scan sampling is performed in the Trifacta Photon running environment.</div> <div>As needed you can force an override of the cache when executing the sample. Data is collected from the source. See <i>Samples Panel</i>.</div>

Recommended Table Size

Although there is no absolute limit, you should avoid executing jobs on tables over several 100 GBs. Larger data sources can significantly impact end-to-end performance.

NOTE: This recommendation applies to all JDBC-based jobs.

Performance

Rule of thumb:

- For a single job with 16 ingest jobs occurring in parallel, maximum expected transfer rate is 1 GB/minute.

Scalability:

- 1 ingest job per source, meaning a dataset with 3 sources = 3 ingest jobs.
- Rule of thumb for max concurrent jobs for a similar edge node:

```
max concurrent sources = max cores - cores used for services
```

- Above is valid until the network becomes a bottleneck. Internally, the above maxed out at about 15 concurrent sources.
- Default concurrent jobs = 16, pool size of 10, 2 minute timeout on pool. This is to prevent overloading of your database.
- Adding more concurrent jobs once network has bottleneck will start slow down all the transfer jobs simultaneously.
- If processing is fully saturated (# of workers is maxed):
 - max transfer can drop to 1/3 GB/minute.
 - Ingest waits for two minutes to acquire a connection. If after two minutes a connection cannot be acquired, the job fails.
- When job is queued for processing:
 - Job is silently queued and appears to be in progress.
 - Service waits until other jobs complete.
 - Currently, there is no timeout for queueing based on the maximum number of concurrent ingest jobs.

Enable

To enable JDBC ingestion and performance caching, the first two of the following parameters must be enabled.

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

Parameter Name	Description
<code>webapp.connectivity.ingest.enabled</code>	Enables JDBC ingestion. Default is <code>true</code> .
<code>feature.jdbcIngestionCaching.enabled</code>	Enables caching of ingested JDBC data. <div>NOTE: <code>webapp.connectivity.ingest.enabled</code> must be set to <code>true</code> to enable JDBC caching.</div>

	When disabled, no caching of JDBC data sources is performed. For more information on caching, see <i>Configure Data Source Caching</i> .
<code>feature.enableLongLoading</code>	When enabled, you can monitor the ingestion of long-loading JDBC datasets through the Import Data page. Default is <code>true</code> . Tip: After a long-loading dataset has been ingested, importing the data and loading it in the Transformer page should perform faster.
<code>feature.enableParquetLongLoading</code>	When enabled, you can monitor the ingestion of long-loading Parquet datasets. Default is <code>false</code> .
<code>longloading.addToFlow</code>	When long-loading is enabled, set this value to <code>true</code> to enable monitoring of the ingest process when large relational sources are added to a flow. Default is <code>true</code> . See <i>Flow View Page</i> .
<code>longloading.addToLibrary</code>	When long-loading is enabled, this feature enables monitoring of the ingest process when large relational sources are added to the library. Default is <code>true</code> . See <i>Library Page</i> .

Configure

In the following sections, you can review the available configuration parameters for JDBC ingest.

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

Configure Ingestion

Parameter Name	Description
<code>batchserver.workers.ingest.max</code>	Maximum number of ingester threads that can run on the Designer Cloud powered by Trifacta platform at the same time.
<code>batchserver.workers.ingest.bufferSizeBytes</code>	Memory buffer size while copying to backend storage. A larger size for the buffer yields fewer network calls, which in rare cases may speed up ingest.
<code>batch-job-runner.cleanup.enabled</code>	Clean up after job, which deletes the ingested data from backend storage. Default is <code>true</code> . NOTE: If JDBC ingestion is disabled, relational source data is not removed from platform backend storage. This feature can be disabled for debugging and should be re-enabled afterward. NOTE: This setting rarely applies if JDBC ingest caching has been enabled.

Logging

Parameter Name	Description
<code>data-service.systemProperties.logging.level</code>	When the logging level is set to <code>debug</code> , log messages on JDBC caching are recorded in the data service log. NOTE: Use this setting for debug purposes only, as the log files can grow quite large. Lower the setting after the issue has been debugged. See Logging below.

Monitoring Progress

You can use the following methods to track progress of ingestion jobs.

- **Through application:** In the Job History page, you can track progress of all jobs, including ingestion. Where there are errors, you can download logs for further review.
 - See *Job History Page*.
 - See Logging below.
- **Through APIs:**
 - You can track status of `jobType=ingest` jobs through the API endpoints.
 - From the above endpoint, get the ingest jobld to track progress.
 - See <https://api.trifacta.com/ee/es.t/index.html#operation/getJobGroup>

Logging

During and after an ingest job, you can download the job logs through the Job History page. Logs include:

- All details including errors
- Progress on ingest transfer
- Record ingestion

See *Job History Page*.

Configure Data Source Caching

Contents:

- *Limitations*
 - *Enable*
 - *Configure*
 - *Configure Storage*
 - *Global or user cache*
 - *Cache sizing*
 - *Logging*
-

This section describes some of the configuration options for the data source caching feature. When data is read from the source, the Designer Cloud powered by Trifacta® platform can populate a global or user-specific cache with ingested objects. These objects can be sourced from:

- JDBC tables, which are ingested as part of running jobs
- Excel data, which must be converted to CSV format and ingested
- PDF table data, which must be converted to CSV format and ingested

After initial ingest, cached objects can be referenced later for faster performance on tasks such as sampling and job execution.

Limitations

- JDBC ingest caching is not supported for Hive.

Enable

To enable JDBC ingestion and performance caching, the following parameter must be enabled.

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

Parameter Name	Description
<code>feature.jdbcIngestionCaching.enabled</code>	<p>Enables caching of ingested JDBC data.</p> <div>NOTE: <code>webapp.connectivity.ingest.enabled</code> must be set to <code>true</code> to enable JDBC caching.</div> <p>When disabled, no caching of JDBC data sources is performed.</p>

Configure

In the following sections, you can review the available configuration parameters for performance caching.

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

Configure Storage

When files are ingested, they are stored in one of the following locations:

- **If caching is enabled:**
 - **If the global datasource cache is enabled:** files are stored in a user-specific sub-folder of the path indicated by the following parameter: `hdfs.pathsConfig.globalDataSourceCache`
 - **If the global cache is disabled:** files are stored in a sub-folder of the output area for each user, named: `/.datasourceCache`.
- **If caching is disabled:** files are stored in a sub-folder within the jobs area for the job group. Ingested files are stored in as `.trifacta` files.

NOTE: Whenever a job is run, its source files must be re-ingested. If two or more datasets in the same job run share the same source, only one copy of the source is ingested.

Additional information is provided below.

Global or user cache

Parameter	Description
<code>datasourceCaching.useGlobalDataSourceCache</code>	<p>When set to <code>true</code>, the platform uses the global data source cache location for storing cached ingest data.</p> <p>NOTE: When global caching is enabled, data is still stored individual locations per user. Through the application, users cannot access the cached objects stored for other users.</p> <p>When set to <code>false</code>, the platform uses the output directory for each user for storing cached ingest data. Within the output directory, cached data is stored in the <code>.datasourceCache</code> directory.</p> <p>NOTE: You should verify that there is sufficient storage in each user's output directory to store the maximum cache size as well as any projected uploaded datasets.</p>
<code>hdfs.pathsConfig.globalDataSourceCache</code>	<p>Specifies the path of the global datasource cache, if it is enabled. Specify the path from the root folder of the backend datastore.</p> <p>Tip: This setting applies to HDFS or other backend datastores.</p>

Cache sizing

Parameter	Description
<code>datasourceCaching.refreshThreshold</code>	<p>The number of hours that an object can be cached. If the object has not been refreshed in that period of time, the next request for the datasource collects fresh data from the source.</p> <p>By default, this value is set to 168 (one week).</p>
<code>datasourceCaching.maxSize</code>	<p>Maximum size in bytes of the datasource cache. This value applies to individual user caches when either global or user-specific caching is enabled.</p>

Logging

Parameter Name	Description
<code>data-service.systemProperties.logging.level</code>	<p>When the logging level is set to <code>debug</code>, log messages on JDBC caching are recorded in the data service log.</p> <p>NOTE: Use this setting for debug purposes only, as the log files can grow quite large. Lower the setting after the issue has been debugged.</p>

See Logging below.

When the logging level is set to `debug` for the data service and caching is enabled, cache messages are logged. These messages include:

- Cache hits and misses
- Cache key generation

Configure Security for Relational Connections

Contents:

- User Security
 - Connection Security Levels
 - Credential Sharing
- Technical Security
 - Encryption Key File
 - SSL
 - Configure long load timeout limits
 - Enable SSO authentication
- Troubleshooting
 - Reading or writing over TLS/SSL fails

You can apply the following Designer Cloud powered by Trifacta® platform features to relational connections to ensure compliance with enterprise practices.

NOTE: These security options apply to external relational connections. For more information configuring security for internal connections to the Trifacta databases, see *Enable SSL for Databases*.

User Security

Connection Security Levels

Connection Security Level	Description
Private	Private connections are created by individuals and are by default accessible only to the individual who created them.
Private and shared	Optionally, they can be shared by individuals with other users. <div>NOTE: If needed, credential sharing can be disabled. See below.</div>
Global	Global connections are either created by administrators or are private connections promoted to global by administrators.

Credential Sharing

By default, users are permitted to share credentials through the application. Credentials can be shared in the following ways:

- A user can create a private connection to a relational database. Through the application, this private connection can be shared with other users, so that they can access the creator's datasets.
- When sharing a flow with another user, the owner of the flow can choose to share the credentials that are necessary to connect to the datasets that are the sources of the flow.

As needed, credential sharing can be disabled.

NOTE: If enterprise policy is to disable the sharing of credentials, collaborators may need to be permitted to store their source data in shared locations.

Tip: Credential sharing can be disabled by individual users when they share a connection. The connection is shared, but the new user must provide new credentials to use the connection.

Steps:

To disable credential sharing at the global level:

1. Login to the application as an administrator.
2. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
3. Locate the following parameter. Set this property to `false`:

```
"webapp.enableCredentialSharing": true,
```

4. Save your changes and restart the platform.

Technical Security

The following features enhance the security of individual and global relational connections.

Encryption Key File

Relational database passwords are encrypted using key files:

- **Passwords in transit:** The platform uses a proprietary encryption key that is invoked each time a relational password is shared among platform services.
- **Passwords at rest:** For creating connections to your relational sources, you must create and reference your own encryption key file. This encryption key is accessing your relational connections from the web application.

This encryption key file must be created and installed on the Trifacta node. For more information, see *Create Encryption Key File*.

SSL

You can enable SSL for any connection by adding the following string to the Connect String Opts field:

```
?ssl=true;
```

Tip: Some connection windows have a Use SSL checkbox, which also works.

Configure long load timeout limits

For long loading relational sources, a timeout is applied to limit the permitted load time. As needed, you can modify this limit to account for larger load times.

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

1. Locate and edit the following parameter:

```
"webapp.connectivity.longLoadTimeoutMillis": 120000,
```

2. Save your changes and restart the platform.

Property	Description
longLoadTimeoutMillis	Max number of milliseconds to wait for a long-loading data source. The default value is 120000 (2 minutes).

For additional relational configuration settings, see *Configure Data Service*.

Enable SSO authentication

Relational connections can be configured to leverage your enterprise Single Sign-On (SSO) infrastructure for authentication. Additional configuration is required. For more information, see *Enable SSO for Relational Connections*.

Troubleshooting

Reading or writing over TLS/SSL fails

Reading or writing over TLS/SSL may fail with an error message in the data service data service log similar to the following:

```
The server selected protocol version TLS11 is not accepted by client preferences [TLS12, SSL20Hello]
```

In this case:

- External libraries referenced by the data service may use TLS/SSL protocols of their own choosing.
- These libraries are included during initialization of the data service.
- The listed protocol (TLSv1.1) is a version of the TLS protocol that is no longer supported.

Solution:

You can configure the platform to override the default protocols supported by Java 8 and to instead use the set of protocols listed in platform configuration.

1. Administrators can apply this configuration change through the *Admin Settings Page* in the application. If the application is not available, the settings are available in `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. When set to `true`, the following parameter instructs the data service to use the protocols listed in Admin Settings page instead. Set this parameter to `true`:

```
"data-service.httpsProtocols.reset": false,
```

Setting	Description
false	(default) Supported HTTPS protocols are defined by Java 8.
true	Supported HTTPS protocols are defined by the Designer Cloud powered by Trifacta platform .

3. Locate the following parameter:

```
"data-service.httpsProtocols.defaultProtocols": "SSLv3,TLSv1,TLSv1.1,TLSv1.2"
```

Tip: You can enter any TLS/SSL protocol supported by Java 8 in the above. Other protocols are likely to cause read/write failures.

4. In this case, you can add the missing protocol to the list, as in the following example:

```
"data-service.httpsProtocols.defaultProtocols": "SSLv3,TLSv1,TLSv1.1,TLSv1.2,TLSv1.1"
```

5. Save your changes and restart the platform.

Enable SSO for Relational Connections

Contents:

- *Limitations*
- *Prerequisites*
- *Configure*
 - *Configure JAAS file and path*
 - *JAAS file*
 - *Specify Kerberos configuration file*
 - *Configure vendor definition file*
- *Example Setup*
- *Use*
 - *Sharing*

This section describes how to enable relational connections to leverage your Hadoop Single Sign-On (SSO) infrastructure. When this feature is enabled and properly configured, users can create relational (JDBC) connections that use SSO that you have already configured.

Connections that were created before this feature is enabled continue to operate as expected without modification.

Limitations

- For this release, this feature applies to SQL Server connections only.
- Cross-realm is not supported. As a result, the SQL Server instance, service principal, and Trifacta® principal must be in the same Kerberos realm.

Prerequisites

- **Kerberos SSO:** You must set up SSO authentication to the Hadoop cluster using Kerberos. This feature uses the global Kerberos keytab. For more information, see *Configure for Kerberos Integration*.

Configure

Configure JAAS file and path

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

Parameter	Description
<code>webapp.connectivity.kerberosDelegateConfigPath</code>	<p>Path on the Trifacta node to the location of the JAAS configuration file required by the DataDirect driver.</p> <div>NOTE: The default location is listed below. You may wish to move this file to a location outside of the Trifacta installation to ensure that the file is not overwritten during upgrades.</div> <p>More information on this file is provided below.</p>

JAAS file

For connections that support Kerberos-delegated authentication, the underlying driver supports a JAAS file in which you can provide environment-specific configuration to the driver. As needed, you can modify this file.

Connection Type	Default path to JAAS file
SQL Server	%(topOfTree)s/services/data-service/build/conf/kerberosdelegate.config

Example JAAS file for SQL Server

Below is an example file, where you must apply the Kerberos global keytab and principal values that are to be used to authenticate to use the Kerberos-delegated connections of this type:

```
trifacta_jaas_config {
  com.sun.security.auth.module.Krb5LoginModule required
  useKeyTab=true
  storeKey=true
  doNotPrompt=true
  keyTab="/absolute/path/to/trifacta_jdbc_sso.keytab"
  principal="<principal_name>";
};

JDBC_DRIVER_01 {
  com.sun.security.auth.module.Krb5LoginModule required debug=false
  useTicketCache=true;
};
```

where:

- **keytab** = the absolute path on the Trifacta node where the Kerberos global keytab is located.
- **principal** = Set to the service principal name of the user's service account in LDAP.

Specify Kerberos configuration file

On the Trifacta node, locate the following file:

```
<root>/etc/krb5.conf
```

If it doesn't exist, create it with the following content, some of which you must specify:

```
[libdefaults]
  default_realm = <my_default_realm>
  forwardable = true # Important that this is set!

[realms]
  <my_default_realm> = {
    kdc = <kdc_domain>
  }

[domain_realm]
  <my_domain> = <my_default_realm>
```

Setting	Description
default_realm	Set this value to your default Kerberos realm.

forwardable	This value must be set to <code>true</code> .
kdc	For each realm that you create, you must create an entry in <code>[realms]</code> . For the <code>kdc</code> entry, apply the KDC domain that the JDBC connection should use.
my_domain	For each domain to which the Kerberos delegation applies, you must create an entry in <code>[domain_realm]</code> . Entries should look like the following: <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <code>example.com = EXAMPLE.COM</code> </div>

Modify the location of the Kerberos configuration file

If you need to move the location of the file from the default one, please complete the following:

Steps:

1. If you haven't already done so, copy the file from its current location to its preferred location.
2. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
3. Specify the path to the new location in the following parameter:

```
"webapp.connectivity.krb5Path": "/etc/krb5.conf";
```

4. Save your changes.

Configure vendor definition file

For each vendor that supports SSO connections you must modify a setting in a configuration file on the Trifacta node. This change can only be applied for vendors that support Kerberized SSO connections.

Steps:

1. On the Trifacta node, navigate to the following directory:

```
/opt/trifacta/services/data-service/build/conf/vendor
```

2. In the `vendor` directory, each JDBC vendor has a sub-directory. Open the vendor directory.
3. Edit `connection-metadata.json`.
4. Locate the `credentialType` property. Set the value to `kerberosDelegate`.
5. Save your changes and restart the platform.
6. When you create your connection, select `kerberosDelegate` from the Credential Type drop-down.

Example Setup

The following example uses the default Kerberos realm to set an SSO connection to a SQL Server instance. This example is intended to demonstrate one way in which you can set up your SSO connections.

Steps:

1. Create the Trifacta service principal:
 - a. Form: `HTTP/serviceprincipal@REAM`

b. Enable this flag: `ok_to_auth_as_delegate`

c. Example:

```
kadmin -q "addprinc -randkey +ok_to_auth_as_delegate HTTP/serviceprincipal"  
kadmin -q "addprinc -randkey +ok_to_auth_as_delegate HTTP/serviceprincipal@REALM"
```

d. For more information on delegation flags, see

https://web.mit.edu/kerberos/krb5-1.12/doc/admin/admin_commands/kadmin_local.html

2. Generate a keytab for the Trifacta service principal.

3. Register the Trifacta service principal for Microsoft Sql Server instance:

a. Enable this flag: `ok_as_delegated`

b. Example:

```
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>:<port>"  
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>:<port>@REALM"  
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>"  
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>@REALM"
```

c. For more information on setting this flag, see

<https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/register-a-service-principal-name-for-kerberos-connections?view=sql-server-2017>

4. Create a linked SQL Server account:

a. Account must have the same name as the end-user principal.

b. Account needs connect permissions at least.

NOTE: If you are using LDAP/AD SSO, you can register all of the above SPNs using AD mechanisms. You do not have to use the delegation flags. Delegation can be managed through the UI for the service account.

Use

When you create a new connection of a supported type, you can select the Kerberos Delegate credentials type. When selected, no username or credentials are applied as part of the connection object. Instead, authentication is determined via Kerberos authentication with the cluster.

- *Microsoft SQL Server Connections*

Sharing

When sharing SSO connections, the credentials for the connection cannot be shared for security reasons. The Kerberos principal for the user with whom the connection is shared is applied. That user must have the appropriate permissions to access any required data through the connection. See *Overview of Sharing*.

Enable OAuth 2.0 Authentication

Contents:

- *Enable*
 - *Enable OAuth 2.0 client creation*
 - *Configure host URL*
 - *Enable Secure Token Service*
 - *Install Secure Token Database*
 - *Create OAuth 2.0 App*
 - *Create OAuth 2.0 Client*
 - *Authenticate OAuth 2.0 Connections*
-

Workspace administrators can enable the use of OAuth 2.0 authentication for creating connections to third-party datastores that support OAuth 2.0 or greater authentication.

OAuth 2.0 is an industry-standard protocol for authorization between systems. In the Designer Cloud powered by Trifacta® platform , it is implemented as a security protocol for access to data sources and publishing destinations. Trifacta administrators can enable users of the product to connect to specified third-party systems through an **OAuth 2.0 client app** that you create in the system, using an **OAuth 2.0 client** reference that is created in the Designer Cloud application .

When enabled and configured, the Designer Cloud application uses the OAuth 2.0 client to create a **secure token**, which is used to authenticate to the third-party system. Internally, the Designer Cloud powered by Trifacta platform leverages the **secure token service** to manage the creation and use of these secure tokens. For OAuth 2.0, this service uses a backing database for storing tokens. **Requirements:**

- **OAuth 2.0 client app:** In the target system, you must create an object called a **client app**, which provides an authentication interface into the system for external connections.
 - You must create one client app for each external system to which you are enabling connectivity.
- **OAuth 2.0 client:** In the Designer Cloud application , you must create at least one configuration object for each client app that you have created.
- Enable the creation of OAuth 2.0 clients in the Designer Cloud application .
- Enable the secure token service, which is used to manage the secure tokens of the Designer Cloud application .
- Install and configure the database used by the secure token service. Installation should happen automatically as part of the normal install or upgrade process.

Details on these requirements are listed below.

Enable

Enable OAuth 2.0 client creation

The ability to create OAuth 2.0 clients in the Designer Cloud application must be enabled. Please verify the following configuration.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Please locate the following setting and set it to `true`:

```
"feature.adminConsole.oauth2ClientsManagement.enabled": true,
```

3. Save your changes.

Configure host URL

When you create an OAuth 2.0 connection, the connection object must pass to the client on the target platform the URL of the Designer Cloud application , so that the client can re-direct queries back to the application after authentication is complete.

Please verify that the following parameter is set to the public value of the host and port number of the Designer Cloud application . It should be in the following form:

```
<http/https>://<host>:<port>
```

where:

- <http/https> = protocol to use to connect
- <host> = host name for external users to access the application
- <port> = port number for external users to access the application. Typically, this value is 3005.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Please verify the following setting is set to the correct value for your environment:

```
"webapp.hostUrl": "https://www.trifacta.example.com:3005",
```

3. Save your changes.

Enable Secure Token Service

OAuth 2.0 requires the use of the secure token service for managing the authentication tokens. For more information, see *Configure Secure Token Service*.

Install Secure Token Database

The secure token service database is installed as part of normal database install or upgrade operations. For more information, see *Install Databases*.

Create OAuth 2.0 App

For each target system, you must create an OAuth 2.0 app in the system, which provides an external interface for the Designer Cloud powered by Trifacta platform .

NOTE: The requirements for creating an OAuth 2.0 app depend on the system. Some example setups are available below. For more information, please see the documentation provided with your target system.

Create OAuth 2.0 Client

Through the Designer Cloud application , you must create an OAuth 2.0 client that connects to the OAuth 2.0 app that you have created.

- In the Admin console, select **OAuth 2.0 Clients**. For more information, see *OAuth 2.0 Clients Page*.
- For more information on creating a client, see *Create OAuth2 Client*.

Authenticate OAuth 2.0 Connections

When you create a connection that uses OAuth 2.0, the specified connection must be authorized to be given access to the datastore. In the Create Connection window, click **Authenticate**.

NOTE: If you modify a connection or the tokens generated under the previous authorization have expired, you must re-authenticate the connection. Edit the connection and click **Re-authenticate**.

Create OAuth2 Client

Contents:

- *Prerequisites*
 - *Enable*
 - *Create OAuth 2.0 App*
 - *Configure*
-

Through the Designer Cloud application , workspace administrators can configure OAuth 2.0 clients to enable connectivity to third-party datastores that support OAuth 2.0 or greater authentication. In the OAuth 2.0 Clients page, click **Register OAuth 2.0 Client**.

Prerequisites

Enable

OAuth 2.0 authentication must be enabled in the Designer Cloud powered by Trifacta® platform . For more information, see *Enable OAuth 2.0 Authentication*.

Create OAuth 2.0 App

Before you create an OAuth Client in the Designer Cloud powered by Trifacta platform , you must create a corresponding Client App in the system with which you are integrating.

Configure

Specify the following properties for your OAuth 2.0 client.

Property	Description
Type	Select the type of client from the drop-down list.
Name	Display name of your OAuth 2.0 client.
Client ID	The client identifier for the OAuth 2.0 app that you created.
Client Secret	The client secret for the OAuth 2.0 app that you created.
Authorization URL	The URL that is used for authorizing to the client app.
Token URL	The token URL for the client app that you created.
Scopes	Scopes are space-delimited strings that can be used to pass parameters to the client app that you created. <div>NOTE: The specific scopes that you can pass depends on the system with which you are integrating.</div>
Access Token Expires In	Number of milliseconds that an access token is permitted to be used to connect to the target OAuth 2.0 app. This value must be set to an integer greater than 0. For more information, please see the documentation for your target system.
Refresh Token Expires In	Number of milliseconds of inactivity that are permitted before an access token is expired. <div>Tip: To create non-expiring tokens, set this value to 0.</div>

For more information, please see the documentation for your target system.

OAuth 2.0 for Google Analytics

Contents:

- *Prerequisites*
 - *Create OAuth 2.0 Client App for Google Analytics*
 - *Enable external user in project*
 - *Create OAuth 2.0 credentials*
 - *Enable API access*
 - *Create OAuth 2.0 Client for Google Analytics*
 - *Create Google Analytics Connection*
-

This section describes the steps to configure the Designer Cloud® application to integrate with Google Analytics using OAuth 2.0 to authenticate.

Prerequisites

- OAuth 2.0 authentication must be enabled in the Designer Cloud powered by Trifacta platform .
- An OAuth 2.0 client is required for Designer Cloud Powered by Trifacta Enterprise Edition only.
- For more information, see *Enable OAuth 2.0 Authentication*.

Create OAuth 2.0 Client App for Google Analytics

Enable external user in project

You must enable external access to the project containing your Google Analytics data.

NOTE: This step configures access through the consent screen for your project. If you have previously completed this step for the project, you can skip this section.

Steps:

1. Navigate to the Google Console for your project: <https://console.cloud.google.com/>.
2. From the left menu, select **APIs & Services > OAuth consent screen**.
3. For User Type, select **External**.
4. Click **Create**.
5. You can provide a logo and name for this client. For example:

Tip: You can use your own logo and product name if preferred.

- a. Right-click the logo in the Designer Cloud application and download it to your desktop. Right-click the image and select **Save As....** Upload it to the consent screen.
 - b. The name of the product can be: Designer Cloud Powered by Trifacta Enterprise Edition.
6. Do not add Scopes or Test Users.
 7. Save your changes.

Create OAuth 2.0 credentials

You must create a set of credentials to use when accessing your Google project.

Steps:

1. From the APIs & Services menu, select **Credentials**.
2. At the top of the screen, click **+CREATE CREDENTIALS**.
3. Select **OAuth client Id**.
4. For Application type, select **Web application**.
5. Fill the values for the following settings:

Setting	Value
Name	Provide a descriptive name. Example: Google_Analytics
Authorized JavaScript origins	Do not add a value for this setting.
Authorized Redirect URIs	Set the value to the following: <code>https://cloud.trifacta.com/oauth2/callback</code>

6. Click **Create**.
7. Retain the values for ClientId and Client Secret. These values must be applied in the Designer Cloud application .

Enable API access

You must enable access to Google Analytics APIs through your project. You can enable one or more of the APIs listed below.

- Google Analytics API: <https://console.cloud.google.com/apis/library/analytics.googleapis.com>
- Google Analytics Reporting API: <https://console.cloud.google.com/apis/library/analyticsreporting.googleapis.com>

Steps:

1. Navigate to listed URL.
2. Click **Enable**.

Create OAuth 2.0 Client for Google Analytics

After the Google Analytics app is created, you must create an OAuth 2.0 client in the Designer Cloud application , which is used to integrate with the OAuth 2.0 connected app that you created above.

NOTE: You must create one OAuth 2.0 client in the Designer Cloud application for each Google Analytics connected app that you wish to use.

Steps:

1. Login to the Designer Cloud application as a workspace administrator.
2. In the lefthand menu, select **User menu > Admin console > OAuth2.0 Clients**.
3. In the OAuth2.0 Clients page, click **Register OAuth2.0 Client**.
4. Specify the new client. Apply the following values:

Setting	Description
Type	Set to <code>google_analytics</code> .
Name	Display name for the OAuth 2.0 client in the Designer Cloud application .
Client ID	Set this value to the Client Id value that you retained from your Google Analytics app.
Client Secret	Set this value to the Client Secret value that you retained from your Google Analytics app.

Authorization URL	Set this value to the following: <div>https://accounts.google.com/o/oauth2/v2/auth</div>
Token URL	Set this value to the following: <div>https://oauth2.googleapis.com/token</div>
Scopes	Please insert the following value: <div>https://www.googleapis.com/auth/drive.readonly</div>
Access Token Expires In	Set this value (in milliseconds) to 3600000 (1 hour).
Refresh Token Expires In	Set the value to 0 (does not expire).

5. To save your OAuth 2.0 client, click **Save**.

For more information, see *Create OAuth2 Client*.

Create Google Analytics Connection

After you have created the two OAuth 2.0 client references, you can create a connection to your Google Analytics data.

NOTE: You must create a separate connection for each OAuth 2.0 client that is available in the Designer Cloud application .

For more information, see *Google Analytics Connections*.

OAuth 2.0 for Google Sheets

Contents:

- *Prerequisites*
 - *Create OAuth 2.0 Client App for Google Sheets*
 - *Enable external user in project*
 - *Create OAuth 2.0 credentials*
 - *Enable API access*
 - *Create OAuth 2.0 Client for Google Sheets*
 - *Create Google Sheets Connection*
-

This section describes the steps to configure the Designer Cloud® application to integrate with Google Sheets using OAuth 2.0 to authenticate.

Prerequisites

- OAuth 2.0 authentication must be enabled in the Designer Cloud powered by Trifacta platform .
- An OAuth 2.0 client is required for Designer Cloud Powered by Trifacta Enterprise Edition only.
- For more information, see *Enable OAuth 2.0 Authentication*.

Create OAuth 2.0 Client App for Google Sheets

Enable external user in project

You must enable external access to the project containing your Google Sheets data.

NOTE: This step configures access through the consent screen for your project. If you have already done this step for the project, you can skip this section.

Steps:

1. Navigate to the Google Console for your project: <https://console.cloud.google.com/>.
2. From the left menu, select **APIs & Services > OAuth consent screen**.
3. For User Type, select **External**.
4. Click **Create**.
5. You can provide a logo and name for this client. For example:

Tip: You can use your own logo and product name if preferred.

- a. Right-click the logo in the Designer Cloud application and download it to your desktop. Right-click the image and select **Save As....** Upload it to the consent screen.
 - b. The name of the product can be: Designer Cloud Powered by Trifacta Enterprise Edition.
6. Do not add Scopes or Test Users.
 7. Save your changes.

Create OAuth 2.0 credentials

You must create a set of credentials to use when accessing your Google project.

Steps:

1. From the APIs & Services menu, select **Credentials**.
2. At the top of the screen, click **+CREATE CREDENTIALS**.
3. Select **OAuth client Id**.
4. For Application type, select **Web application**.
5. Fill the values for the following settings:

Setting	Value
Name	Provide a descriptive name. Example: Google_Analytics
Authorized JavaScript origins	Do not add a value for this setting.
Authorized Redirect URIs	Set the value to the following: <code>https://cloud.trifacta.com/oauth2/callback</code>

6. Click **Create**.
7. Retain the values for ClientId and Client Secret. These values must be applied in the Designer Cloud application .

Enable API access

You must enable API access to your project.

Steps:

1. To enable the Google Sheets API, navigate to the following URL:
<https://console.cloud.google.com/apis/library/sheets.googleapis.com>
2. Click **Enable**.
3. To use Google Sheets, you must also enable the Google Drive API. Navigate to the following URL:
<https://console.cloud.google.com/apis/library/drive.googleapis.com>
4. Click **Enable**.

Create OAuth 2.0 Client for Google Sheets

After the Google Sheets app is created, you must create an OAuth 2.0 client in the Designer Cloud application , which is used to integrate with the OAuth 2.0 connected app that you created above.

NOTE: You must create one OAuth 2.0 client in the Designer Cloud application for each Google Sheets c onnected app that you wish to use.

Steps:

1. Login to the Designer Cloud application as a workspace administrator.
2. In the lefthand menu, select **User menu > Admin console > OAuth2.0 Clients**.
3. In the OAuth2.0 Clients page, click **Register OAuth2.0 Client**.
4. Specify the new client. Apply the following values:

Setting	Description
Type	Set to <code>google_sheets</code> .
Name	Display name for the OAuth 2.0 client in the Designer Cloud application .
Client ID	Set this value to the Client Id value that you retained from your Google Sheets app.
Client Secret	Set this value to the Client Secret value that you retained from your Google Sheets app.

Authorization URL	Set this value to the following: <code>https://accounts.google.com/o/oauth2/v2/auth</code>
Token URL	Set this value to the following: <code>https://oauth2.googleapis.com/token</code>
Scopes	Please insert the following value: <code>https://www.googleapis.com/auth/drive.readonly</code>
Access Token Expires In	Set this value (in milliseconds) to 3600000 (1 hour).
Refresh Token Expires In	Set the value to 0 (does not expire).

5. To save your OAuth 2.0 client, click **Save**.

For more information, see *Create OAuth2 Client*.

Create Google Sheets Connection

After you have created the two OAuth 2.0 client references, you can create a connection to your Google Sheets data.

NOTE: You must create a separate connection for each OAuth 2.0 client that is available in the Designer Cloud application .

For more information, see *Google Sheets Connections*.

OAuth 2.0 for Salesforce

Contents:

- *Prerequisites*
 - *Create OAuth 2.0 Client App in Salesforce*
 - *Scopes for Salesforce*
 - *Create OAuth 2.0 Client for Salesforce*
 - *Create Salesforce Connection*
-

This section describes the steps to configure the Designer Cloud® application to integrate with your Salesforce deployment using OAuth 2.0 to authenticate.

Prerequisites

OAuth 2.0 authentication must be enabled in the Designer Cloud powered by Trifacta platform . For more information, see *Enable OAuth 2.0 Authentication*.

Create OAuth 2.0 Client App in Salesforce

In Salesforce, you must create the connected app through which the Designer Cloud application uses OAuth 2.0 to access and connect to your Salesforce data.

Steps:

1. **Login:** Log in to the Salesforce account in which you want the OAuth 2.0 app to be created.
2. In the top bar, click **Setup**.
3. In the left nav bar, search for: `apps`. Then, navigate to **Create > Apps**.
4. **Create connected app:** In the Connected Apps section, click **New**.
 - a. To create a connected app, please complete the listed fields with the appropriate information. Some specifics:

Field	Description
Connected App Name	Display name of the app. Suggested: Trifacta application
API Name	Please add the value for Connected App Name here.
Contact Email	Add a valid contact email address.
Logo image URL	(optional) Upload an app logo as needed.
Enable OAuth Settings	Select this option.
Callback URL	Please provide a URL in the following format: <code>https://cloud.trifacta.com/oauth2/callback</code> This value may or may not include a port number.
Selected OAuth Scopes	Please select the following scopes:1. <code>api2.refresh_token</code>
Require secret for web server flow	Select this option.

- b. At the bottom of the screen, click **Save** to save the connected app.
5. **Configure policies:** In the left nav bar, select **Manage > Connected apps**.

- a. Then, click the **Edit Policies** button.
 - b. In the Edit Policies screen, click the **Manage** button.
 - c. Under Session Policies, select the Timeout Value. Set this value to 24 hours.
 - d. Click **Save** to save your connected app.
6. **Retain values:** Your Salesforce connected app configuration is complete. Please acquire the following information from the app listing in Salesforce. These parameter values are needed for creating the OAuth 2.0 client in the Designer Cloud application :

Parameter	Description
Consumer Key	This value is used as the Client Id in Designer Cloud application . Select Click to reveal to display.
Consumer Secret	This value is used as the Client Secret in Designer Cloud application . Select Click to reveal to display.
Selected OAuth Scopes	Acquire this values. Unless otherwise specified, these values should include:1. api2. refresh_token
Access token expires in	Navigate to Manage > Edit Policies . Typically, this value in milliseconds is set to 1 hour (3600000 milliseconds). For more information, see https://help.salesforce.com/articleView?id=connected_app_manage_session_policies.htm&type=5 .

7. Save any changes to the connected app.

Scopes for Salesforce

The following scopes are required in the connected app for the Designer Cloud application to access Salesforce:

Scope	Description
api	(required) Provides REST API access to Salesforce.
refresh_token	(required) This token allows the OAuth 2.0 client to refresh the connection with Salesforce without user interaction.

Create OAuth 2.0 Client for Salesforce

After the Salesforce connected app is created, you must create an OAuth 2.0 client in the Designer Cloud application , which is used to integrate with the OAuth 2.0 connected app that you created above.

NOTE: You must create one OAuth 2.0 client in the Designer Cloud application for each Salesforce connected app that you wish to use.

Steps:

1. Login to the Designer Cloud application as a workspace administrator.
2. In the lefthand menu, select **User menu > Admin console > OAuth2.0 Clients**.
3. In the OAuth2.0 Clients page, click **Register OAuth2.0 Client**.
4. Specify the new client. Apply the following values:

Setting	Description
Type	Set to <code>salesforce</code> .
Name	Display name for the OAuth 2.0 client in the Designer Cloud application .
Client ID	Set this value to the Consumer Key value in your Salesforce connected app.
Client Secret	Set this value to the Consumer Secret value in your Salesforce connected app.
Authorization URL	Set this value to the following: <div></div>

	<code>https://login.salesforce.com/services/oauth2/authorize</code>
Token URL	Set this value to the following: <code>https://login.salesforce.com/services/oauth2/token</code>
Scopes	Insert the scopes you specified as a space-separated list.
Access Token Expires In	Set this value to the corresponding value in your Salesforce connected app. See above.
Refresh Token Expires In	Set this value to the number of milliseconds after which the refresh token expires. Set the value to 0 (does not expire).

5. To save your OAuth 2.0 client, click **Save**.

For more information, see *Create OAuth2 Client*.

Create Salesforce Connection

After you have created the two OAuth 2.0 client references, you can create a connection to your Salesforce data.

NOTE: You must create a separate connection for each OAuth 2.0 client that is available in the Designer Cloud application .

For more information, see *Salesforce Connections*.

OAuth 2.0 for SharePoint

Contents:

- *Prerequisites*
- *Create OAuth 2.0 Client App in Azure*
- *Create OAuth 2.0 Client for SharePoint*
- *Create SharePoint Connection*

Configure the Designer Cloud application to integrate with your SharePoint deployment using OAuth 2.0 to authenticate.

Prerequisites

- OAuth 2.0 is supported for SharePoint Online, which is available on Microsoft Azure.
- OAuth 2.0 authentication must be enabled in the Designer Cloud powered by Trifacta platform . For more information, see *Enable OAuth 2.0 Authentication*.

Create OAuth 2.0 Client App in Azure

In the Azure console, you must create the client app.

Steps:

1. Login to Azure: <https://portal.azure.com/#home>
2. Select **Azure Active Directory > App Registrations > New registration**.
3. Specify the following fields:

Field	Description
Name	The name of the app.
Supported Account Types	Select either: <ul style="list-style-type: none">• Single tenant• Multi tenant
Redirect URI	Specify the URL in the following format, depending your project or workspace login: <div><code>https://<Login_URL>/oauth2/callback</code></div> <div>Example:</div> <div><code>https://cloud.trifacta.com/oauth2/callback</code></div>

4. To create the new registration, click **Register**.
5. **Overview tab:**

NOTE: Copy the value for the Application (client) Id. This value must be applied in the Designer Cloud application .

6. **Branding tab:** (optional) Specify the following fields:

Field	Description
Name	The name of the app should match the value you specified previously.

Logo	Upload a preferred logo for the app.
Homepage URL	Set this value to the following: <code>https://trifacta.com</code>
Terms of Service URL	Set this value to the following: <code>https://www.trifacta.com/terms-conditions/</code>
Privacy Policy	Set this value to the following: <code>https://www.trifacta.com/privacy-policy/</code>
Publisher Domain	Set this value to the following: <code>trifacta.com</code>
MPN ID	If you are a Microsoft Partner, you can specify your MPN ID.

7. **Certificates & Secrets tab:** To create a new client secret:

- Select **New client secret**.
- Set the Expires option to *Never*.
- Select **Add**.

NOTE: Copy the Value field. This value is the client secret and must be applied in the Designer Cloud application .

8. **API Permissions tab:**

- Select **Add a permission**.
- Select **Sharepoint**.
- Specify the permissions to enable for the client app.

NOTE: To be able to read from SharePoint Online, the following permission is required at a minimum: `AllSites.Manage`.

9. Save your changes.

Create OAuth 2.0 Client for SharePoint

After the SharePoint client app is created, you must create an OAuth 2.0 client in the Designer Cloud application , which is used to integrate with the OAuth 2.0 Client app that you created in Azure.

Steps:

- Login to the Designer Cloud application as a workspace administrator.
- In the lefthand menu, select **User menu > Admin console > OAuth 2.0 Clients**.
- In the OAuth 2.0 Clients page, click **Register OAuth 2.0 Client**.
- Specify the fields for the new client:

Field	Description
Type	sharepoint

Name	Enter a name for the client.
Client ID	Paste the value of the Application (client) Id that was generated when you created the client app in Azure.
Client Secret	Paste the value of the client secret that was generated when you created the client app in Azure.
Authorization URL	<p>Single tenant:</p> <div>https://login.microsoftonline.com/<tenant_identifier>/oauth2/v2.0/authorize</div> <p>Multi-tenant:</p> <div>https://login.microsoftonline.com/common/oauth2/v2.0/authorize</div>
Token URL	<p>Single tenant:</p> <div>https://login.microsoftonline.com/<tenant_identifier>/oauth2/v2.0/token</div> <p>Multi-tenant:</p> <div>https://login.microsoftonline.com/common/oauth2/v2.0/token</div>
Scopes	<p>Set the scopes to the following values:</p> <div>NOTE: Individual scopes must be separated by a space.</div> <div>https://<your_sharepoint_domain>/AllSites.Manage offline_access openid</div>
Access Token Expires In	Set this value to the following: 3599999.
Refresh Token Expires In	Set this value to the following: 7775999999.

5. To save your OAuth 2.0 client, click **Save**.

For more information, see *Create OAuth2 Client*.

Create SharePoint Connection

After you have created the two OAuth 2.0 client references, you can create a connection to your SharePoint data.

NOTE: You must create a separate connection for each OAuth 2.0 client that is available in the Designer Cloud application .

For more information, see *SharePoint Connections*.

OAuth 2.0 for Snowflake

Contents:

- *Prerequisites*
- *Create OAuth 2.0 Client App in Snowflake*
- *Create OAuth 2.0 Client for Snowflake*
 - *Scopes for Snowflake*
- *Create Snowflake Connection*
- *Troubleshooting*
 - *"Invalid consent request" error*

Configure the Designer Cloud application to integrate with your Snowflake deployment using OAuth 2.0 to authenticate.

Prerequisites

OAuth 2.0 authentication must be enabled in the Designer Cloud powered by Trifacta platform . For more information, see *Enable OAuth 2.0 Authentication*.

Create OAuth 2.0 Client App in Snowflake

In your Snowflake console, you must create the client app, which includes execution of several SQL statements.

NOTE: You must have the ACCOUNTADMIN role to create the client app.

In Snowflake, this object is called a **security integration**. For more information, see <https://docs.snowflake.com/en/sql-reference/sql/create-security-integration.html>.

Steps:

1. Login to the Snowflake console as an account admin.
2. Click **Worksheets**.
3. For your role, select **ACCOUNTADMIN**.
4. Paste the following command in the worksheet and modify its parameters:

```
CREATE [ OR REPLACE ] SECURITY INTEGRATION [IF NOT EXISTS]
  <NAME>
  TYPE = OAUTH
  OAUTH_CLIENT = CUSTOM
  OAUTH_CLIENT_TYPE = 'CONFIDENTIAL'
  OAUTH_REDIRECT_URI = '<URI>'
  ENABLED = TRUE
  OAUTH_ALLOW_NON_TLS_REDIRECT_URI = FALSE
  [PRE_AUTHORIZED_ROLES_LIST = ( '<role_name_1>' [ , '<role_name_2>' , ... ] ) ]
  [ BLOCKED_ROLES_LIST = ( '<role_name_3>' [ , '<role_name_4>' , ... ] ) ]
  OAUTH_ISSUE_REFRESH_TOKENS = TRUE
  OAUTH_REFRESH_TOKEN_VALIDITY = 7776000 (90 Days)
  [ NETWORK_POLICY = '<network_policy>' ]
  [ COMMENT = '<Description of your Integration>' ]
```

Parameter	Description
<NAME>	Name of the integration. Example: OAuth 2.0 Client
<URI>	Callback URI of the Designer Cloud powered by Trifacta platform . https://cloud.trifacta.com/oauth2/callback
PRE_AUTHORIZED_ROLES_LIST	A comma-separated list of Snowflake roles that do not need user consent when accessing Snowflake. The roles SECURITYADMIN and ACCOUNTADMIN cannot be included in this list. <div> Tip: The roles in this list should match up with the roles that are scoped in the OAuth 2.0 client in the Designer Cloud application . In the client, you can specify the Snowflake roles that are permitted to use the client for authentication. Roles that are scoped for access that are not in this list must consent to access Snowflake after login. In some use cases, such as API access or scheduled executions, this can be problematic. </div>
BLOCKED_ROLES_LIST	A comma-separated list of Snowflake roles that cannot explicitly consent to use when accessing Snowflake. The roles SECURITYADMIN and ACCOUNTADMIN are included by default in this list. If you need to remove either of those roles, please contact Snowflake Support.
<NETWORK_POLICY>	(Optional) Provide the identifier for any applicable Snowflake network policy.
<COMMENT>	(Optional) Add a comment if needed.

- Run the above command. The security integration is created.
- Paste the following command and run it to acquire the following information: Client ID, Authorization URL, Token URL, and Refresh Token Expires In, where <NAME> Is the name you provided above:

```
DESC integration <NAME>
```

Retain the values for the following parameters. You must apply these parameters to the OAuth 2.0 client that you create in the Designer Cloud application :

Snowflake parameter	Designer Cloud application Client parameter
OAUTH_CLIENT_ID	Client Id
OAUTH_AUTHORIZATION_ENDPOINT	Authorization URL
OAUTH_TOKEN_ENDPOINT	Token URL
OAUTH_REFRESH_TOKEN_VALIDITY	Refresh Token Expires In

- Paste the following command and run it to acquire the client secret, where <NAME> Is the name you provided above:

```
SELECT SYSTEM$SHOW_OAUTH_CLIENT_SECRETS( ' <NAME> ' )
```

Retain the values for the following. You must apply these parameters to the OAuth 2.0 client that you create in the Designer Cloud application :

Snowflake parameter	Designer Cloud application Client parameter
OAUTH_CLIENT_SECRET	Client Secret

- Save your changes.

Create OAuth 2.0 Client for Snowflake

After the Snowflake client app is created, you must create an OAuth 2.0 client in the Designer Cloud application, which is used to integrate with the OAuth 2.0 Client app (security integration) that you created above.

NOTE: You must create one OAuth 2.0 client in the Designer Cloud application for each Snowflake role that you wish to use. See "Scopes" below for more information.

Steps:

1. Login to the Designer Cloud application as a workspace administrator.
2. In the lefthand menu, select **User menu > Admin console > OAuth 2.0 Clients**.
3. In the OAuth 2.0 Clients page, click **Register OAuth 2.0 Client**.
4. Specify the new client.
 - a. For the Type value, select `snowflake`.
 - b. You must apply the values listed in the previous section to your client object.
 - c. For more information on Scopes, see "Scopes for Snowflake" below.
 - d. Access Token Expires in: 600000

NOTE: The value of 600000 is required for Snowflake.

5. To save your OAuth 2.0 client, click **Save**.

For more information, see *Create OAuth2 Client*.

Scopes for Snowflake

Scopes are space-delimited strings that are passed from the client to the client app as part of the authentication process.

The following scope must be specified as part of your Snowflake client definition:

```
refresh_token session:role:<role_name>
```

Scope	Description
refresh_token	(required) Snowflake session tokens have a short duration. By adding this scope, a refresh token is issued for the session. This token allows the OAuth 2.0 client to refresh the connection with Snowflake without user interaction.
role:<role_name>	<p>(optional) The Snowflake role for which you wish to access its databases, schemas, and tables. If this value is not provided, then the default role is used.</p> <p>NOTE: Only one role can be specified per client. This role must provide access to the databases, schemas, and objects that you wish to make accessible through this client.</p> <p>NOTE: The value for <role_name> is case-sensitive, unless you specified the role in quotes when creating it. For more information, see https://docs.snowflake.com/en/user-guide/oauth-custom.html#scope.</p>

Create Snowflake Connection

After you have created the two OAuth 2.0 client references, you can create a connection to your Snowflake databases.

NOTE: You must create a separate connection for each OAuth 2.0 client that is available in the Designer Cloud application .

For more information, see *Snowflake Connections*.

Troubleshooting

The following may occur when trying to connect to Snowflake databases using OAuth 2.0.

"Invalid consent request" error

If you receive an invalid consent request error, then the user that is passed for OAuth 2.0 authorization does not have access to the role that is referenced in the corresponding OAuth 2.0 client that you created in the Designer Cloud application .

You can do one of the following:

- Specify a different user in the connection.
- Create a new OAuth 2.0 client in the Designer Cloud application which is scoped for a role that the database user has.

NOTE: This new role must also be authorized to use the security integration within Snowflake.

Configure Connectivity for Amazon RDS

This section outlines additional configuration that may be required to create connections between the Designer Cloud powered by Trifacta® platform and your relational instances hosted on Amazon RDS.

Configure SSL Connections

You can configure your connection to use SSL for interactions with your Amazon RDS database.

Database server configuration: On the database side, you must disable SSL certificate validation. For more information, see <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.SSL.html>.

Designer Cloud application : Since you cannot upload your own SSL certificate to the Designer Cloud powered by Trifacta platform , you must perform the following steps to disable SSL validation. These steps must be applied to any database connection in the Designer Cloud application to an Amazon RDS database instance.

Steps:

1. In the Connections page, select the Amazon RDS database connection that you wish to modify.
2. In the Edit Connection window:
 - a. Select the Enable SSL checkbox.
 - b. For the Connect String Options:
 - i. Find the appropriate connect string to disable client certification validation. See the documentation that was provided with your database distribution.
 - ii. Add that value to any existing value for the Connect String Options.
 - c. Test the connection.
3. Save your changes.

When the above is performed, the Designer Cloud application is no longer expecting a valid certificate and can use SSL to communicate.

Configure Type Inference

Contents:

- *Configure Type Inference for Schematized Sources*
 - *Enable*
 - *Configure Load Limits for Inference*
 - *Use*
 - *Define for individual connections*
 - *Specify on dataset import*
 - *Configure Type Inference in the Data Grid*
 - *Type Inference on Export*
-

By default, the Designer Cloud powered by Trifacta® platform applies its own type inference to datasets when they are imported and again when new steps are applied to the data. This section provides information on how you can configure where type inference is applied in the platform.

Data types are inferred by the Designer Cloud powered by Trifacta platform when:

- Imported datasets are originally loaded.
- A new transformation step is added in a recipe.
- Non-inferred types are imported as String type.

Tip: You can use the Change Column Type transformation to override the data type inferred for a column. However, if a new transformation step is added, the column data type is re-inferred, which may override your specific typing. You should consider applying Change Column Type transformations as late as possible in your recipes.

For more information on how the Designer Cloud powered by Trifacta platform applies data types to specific sources of data on import, see *Type Conversions*.

Configure Type Inference for Schematized Sources

Optionally, you can choose to disable type inference for schematized sources. A **schematized source** includes column data type information as part of the object definition. The following schematized sources are supported for import into the Designer Cloud powered by Trifacta platform :

- All JDBC sources

NOTE: You cannot disable type inference for Oracle sources. This is a known issue.

- Hive
- Redshift
- Avro file format

Enable

Type inference on schematized sources	Setting	Behavior
Enabled	"webapp.connectivity.disableRelationalTypeInference": false,	<p>All imported datasets from schematized sources are automatically inferred by the type system in the Designer Cloud powered by Trifacta platform .</p> <p>The inferred data types may be different from those in the source. When the dataset is loaded, data types can be applied to individual columns through the application.</p> <p>Users can apply overrides for:</p> <ul style="list-style-type: none">• Individual connections• Individual datasets at time of import
Disabled	"webapp.connectivity.disableRelationalTypeInference": true,	<p>For schematized data sources, type inference is not automatically inferred by Designer Cloud powered by Trifacta platform .</p> <p>Data type information is taken from the source schema and applied where applicable to the dataset. If there is no corresponding data type in the Designer Cloud powered by Trifacta platform , the data is imported as String type.</p> <p>Users can apply overrides for:</p> <ul style="list-style-type: none">• Individual connections• Individual datasets at time of import

Please perform the following configuration change to disable type inference of schematized sources at the global level.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Change the following configuration setting to `true`:

```
"webapp.connectivity.disableRelationalTypeInference": false,
```

3. Save your changes.

Configure Load Limits for Inference

When a dataset is imported into the Designer Cloud application , a volume of data is read from the source, up to the parameterized limits below. These limits define the maximum size of the data read for:

- **Split row inference:** data read for determining where each row ends in the dataset.
- **Type inference:** data read for determining the data types of each column.

Tip: You can raise these limits gradually if you are noticing issues with either data inference or row splits. Raising these values significantly can impact load performance in the Transformer page.

Parameter	Description
-----------	-------------

webapp. loadLimitForSplitInference	Maximum number of bytes to be read from an imported dataset for initial inference for splitting rows. Default value is 20000.
webapp. loadLimitForTypeInference	Maximum number of bytes to be read from an imported dataset for initial inference of column data types. Default value is 524288.

Use

In the application, type inference can be applied to your imported data through the following mechanisms.

Define for individual connections

You can specify individual connections to apply or not apply Trifacta type inference when the connection is created or edited.

NOTE: When Default Column Data Type Inference is disabled for an individual connection, Trifacta type inference can still be applied on import of individual datasets.

For more information, see *Create Connection Window*.

Specify on dataset import

When type inference has been disabled globally for schematized sources, you can choose to enable or disable it for individual source import.

Tip: To compare how data types are imported from the schematized source or when applied by the Designer Cloud powered by Trifacta platform, you can import the same schematized source twice. The first instance of the source can be imported with type inference enabled, and the second can be imported with it disabled.

In the Import Data page, click **Edit Settings** on the data source card.

For more information, see *Import Data Page*.

Configure Type Inference in the Data Grid

Type inference is automatically enabled in the data grid. It cannot be disabled.

Tip: You can override the Trifacta data type by applying a Change Column Type transformation.

When a new transformation step is applied, each column is re-inferred for its Trifacta data type.

Type Inference on Export

When you generate results, the current data types in the data grid are applied to the generated results.

If the publishing destination is a schematized environment, the generated results are written to the target environment based on the environment type. These data type mappings cannot be modified.

For more information on output types, see *Type Conversions*.

Troubleshooting Relational Connections

Contents:

- *Problem - Unable to access customer encryption key*
 - *Solution*
 - *Problem - Retrieving sample data for large relational tables is very slow*
 - *Solution*
-

This section provides generic information on how to troubleshoot your relational connections.

Problem - Unable to access customer encryption key

When trying to create, edit, or test a relational connection, you may receive the following error message:

```
400 - Encryption Key Error. Please Contact Administrator:
Unable to access customer encryption key.
```

You are unable to access the relational source.

Solution

The encryption keyfile is missing from the Trifacta® deployment, or the keyfile has been moved without updating the platform of the new location.

You must create and deploy this keyfile, which is required for ensuring that encrypted usernames and passwords are used in relational connections.

NOTE: This keyfile must be created and deployed before any relational connections are created. Deployment requires access to the file system on the Trifacta node.

After you have deployed the keyfile, you must configure the platform to point to its location. A platform restart is not required.

For more information, see *Relational Access*.

Problem - Retrieving sample data for large relational tables is very slow

In some cases, you may experience slow performance in reading from database tables, or previews of large imported datasets are timing out.

Solution

In these cases, you can experiment with the number of records that are imported per database read. By default, this value is 25000.

To improve performance, you can modify the following setting.

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

```
"data-service.sqlOptions.limitedReadStreamRecords": 25000,
```

To improve performance, you can try lowering this limit incrementally. Avoid raising this limit over 100000, which can overwhelm the browser.

Supported Connection Credential Types

Contents:

- *API Key*
- *API Key with Token*
- *Azure Token SSO*
- *AWS*
- *AWS Key/Secret*
- *Basic*
- *Basic app*
- *Basic with app token*
- *conf*
- *HTTP Header-Based Authentication*
- *HTTP Query-Based Authentication*
- *IAM DB User*
- *IAM Role Arn*
- *Kerberos Delegate*
- *Kerberos Impersonation*
- *Key/Secret*
- *No Authentication*
- *OAuth 2.0*
- *Password*
- *Security Token*
- *SSH Key*
- *SSH Tunneling Basic*
- *SSH Tunneling SSH Key*
- *Transaction Key*
- *User with API Token*
- *Reference Information*
 - *Connections by Credential Type*
 - *API References*

This section contains general reference information on the credential types that are supported for use in connections from the Designer Cloud application . A **credential type** defines the authentication or account information that must be provided to the authenticating application.

NOTE: Some credential types may not be available in your product edition.

API Key

This credential type requires generation of an API key within the target application. This key must be inserted as part of the connection definition in the Designer Cloud application .

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["apiKey"]
```

API Key with Token

This credential type requires an API key generated by the target application, as well as an access token tied to the API key.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["apiKeyWithToken"]
```

Azure Token SSO

Connect to Azure-hosted resources using the Azure Single Sign On (SSO) token for the authenticating user.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["azureTokenSso"]
```

AWS

AWS-specific credentials. Used for Redshift connections.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["aws"]
```

AWS Key/Secret

These AWS-specific credentials use a key/secret combination to authenticate to AWS systems, such as Amazon Dynamo DB and Amazon Athena.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["awsKeySecret"]
```

Basic

A simple username/password can be provided to the authenticating application.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["basic"]
```

Basic app

The basic app credential type requires that a private app be created in the target application. Access through this app needs an AppId and Password combination.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["basicApp"]
```

Basic with app token

This basic authentication mechanism requires three pieces of information: Username, Password and Application Token. All of these are available in through the target application.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["basicWithAppToken"]
```

conf

For this credential type, the connection credentials are stored in `trifacta-conf.json`, a JSON configuration file stored on the node hosting the product.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["conf"]
```

HTTP Header-Based Authentication

Used for REST API connections, these credentials are submitted as key/value pairs in the HTTP request.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["httpHeaderBasedAuth"]
```

HTTP Query-Based Authentication

Used for REST API connections, these credentials are submitted as key/value pairs in URL query parameters.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["httpQueryBasedAuth"]
```

IAM DB User

This credential type leverages an IAM role to access Amazon Redshift databases. The IAM role must be specified as part of the connection definition.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["iamDbUser"]
```

IAM Role Arn

This credential type uses an IAM role to access external S3 buckets, which are not defined as part of the base storage layer.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["iamRoleArn"]
```

Kerberos Delegate

Connection uses the Kerberos-delegated principal to connect to a relational database. No credentials are submitted as part of the connection definition. This method requires additional configuration.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["kerberosDelegate"]
```

Kerberos Impersonation

Connection uses the Kerberos impersonation principal for the user to connect to the database. No credentials are submitted as part of the connection definition.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["kerberosImpersonation"]
```

Key/Secret

When accessing an external S3 bucket, you can apply key-secret combinations as part of your connection definition. This authentication mechanism consists of an AWS Access Key ID and an AWS Access Secret ID.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["keySecret"]
```

No Authentication

Some connection types do not require credentials to be submitted to them.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["noAuth"]
```

OAuth 2.0

OAuth 2.0 credentials can be used to connect a client in the Designer Cloud application to the client app created in the target system.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["oauth2"]
```

NOTE: Additional configuration may be required to enable this credential type for a specific connection type.

Password

A single password value is required for authentication.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["password"]
```

Security Token

This credential type requires the insertion of a single security token as part of the connection definition. This security token must be generated from the targeted application.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["securityToken"]
```

SSH Key

Used for SFTP connections, this credential type requires that you insert an SSH key generated from the host server of the FTP site.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["sshKey"]
```

SSH Tunneling Basic

For SSH tunneling connectivity, you can use a simple username and password set of credentials. This credential type can be applied to various connection types.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["sshTunnelingBasic"]
```

SSH Tunneling SSH Key

For SSH tunneling connectivity, you can use a username and SSH key as a set of credentials. This credential type can be applied to various connection types.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["sshTunnelingSshKey"]
```

Transaction Key

This credential type uses a Login ID and Transaction Key to authenticate.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": ["transactionKey"]
```

User with API Token

This credential type requires a user identifier and an API token associated with that user to authenticate to the server.

Trifacta API attribute:

When creating a connection via API, the following attribute and value must be inserted as part of the connection definition:

```
"credentialType": [ "userWithApiToken" ]
```

Reference Information

Connections by Credential Type

Credential Type	Connection Type
apiKey	Airtable , Freshdesk , HubSpot , Mailchimp , SendGrid
apiKeyWithToken	Trello
awsKeySecret	Amazon Athena , Amazon DynamoDB
azureTokenSource	Azure SQL Database
basic	Alloy DB , MariaDB on Amazon RDS , MySQL on Amazon RDS , Oracle DB on Amazon RDS , PostgreSQL on Amazon RDS , SQL Server on Amazon RDS , Apache Impala , Azure SQL Database , Cassandra DB , MySQL on Google Cloud SQL , PostgreSQL on Google Cloud SQL , SQL Server on Google Cloud SQL , Cockroach DB , DB2 , Denodo , Greenplum , IBM DB2 , REST API , Jira by Atlassian , Magento , MariaDb , MongoDB , MongoDB Atlas , MySQL , Oracle Database , PostgreSQL , Presto , SAP HANA , ServiceNow , SFTP , SharePoint , Snowflake , Snowflake JDBC (Private Preview) , Splunk , Azure Synapse Analytics (Formerly Microsoft SQL DW) , Microsoft SQL Server , Tableau Server , Teradata , Trino , Workday , Zendesk
basicApp	Shopify
basicWithAppToken	Quickbase
conf	Databricks , Amazon Glue , Hive
httpHeaderBasedAuth	\$strConnectionType
iamDbUser	Amazon Redshift
iamRoleArn	Amazon Redshift
keySecret	External Amazon S3
noAuth	REST API , Presto , Trino
oauth2	Adobe Analytics , Asana , Microsoft Advertising , Microsoft Dataverse , Denodo , Microsoft Dynamics 365 Sales , Microsoft Dynamics 365 Sales , Exact Online , Facebook Ads , Google Ads , Google Analytics , Google BigQuery JDBC (Private Preview) , Google Calendar , Google Contacts , Google Data Catalog , Google Spanner , Google Sheets , Instagram Ads , LinkedIn Ads , Marketo , NetSuite , Pinterest , QuickBooks Online , Salesforce , SharePoint , Smartsheet , Snowflake , SurveyMonkey , Workday , Xero , YouTube Analytics , Zoho CRM
password	Redis
securityToken	SalesForce (Deprecated) , Salesforce
sshKey	SFTP
transactionKey	Authorize.Net

API References

In the request and response for actual connections, the attribute `credentialTypes` is used as a String value:

```
{
  "id": 37,
  "host": "postgres.example.com",
  "port": 5432,
  "vendor": "postgres",
  "params": {
    "connectStrOpts": "",
    "database": "mydb"
  },
  "ssl": false,
  "vendorName": "postgres",
  "name": "Postgres20200417182437287",
  "description": "",
  "type": "jdbc",
  "isGlobal": false,
  "credentialType": "basic",
  "credentialsShared": false,
  "uuid": "myUniqueId",
  "disableTypeInference": false,
  "createdAt": "2020-04-17T18:25:04.518Z",
  "updatedAt": "2020-04-17T18:25:04.530Z",
  ...
}
```

Connect to Data

Contents:

- *Locate Connections*
 - *Use Connections*
 - *Read-Only*
 - *Write*
 - *Create Connection*
 - *Delete Connection*
-

When you import data into the Designer Cloud powered by Trifacta® platform, you are creating a reference to a source of data; the source is never touched. When the data is required for use, the Designer Cloud powered by Trifacta platform reads a sample of the source data into the application for your use. Data is read into the application through an object called a **connection**.

The following are the supported types of connection for the product:

- **Upload/Download:** You can upload data directly from your local desktop. You can also save it locally on export.
- **Base storage layer:** Your deployed instance of the product is connected to a base storage layer, where you can read sources and write your results.
- **Relational sources:** You can read from database tables into the product.

Locate Connections

You already have a set of connections that you can use. Connections can be either read-only or read-write.

1. In the Home page, click the Connections icon in the left nav bar.
2. The currently available Connections is displayed.

In the Import Data page, your list of available connections is displayed in the left nav bar.

Use Connections

Read-Only

1. In the Import Data page, select one of the available connections.
2. Navigate through the connection to select the asset to import.
3. Select the object and click **Open**.
4. In the Import Data page, review the settings of the asset in the card in the right panel. Make updates as needed.

Write

You write results through a connection by specifying a set of settings.

1. In the Run Job page, click **Add Publishing Action**.
2. In the left nav bar, select the connection.
3. Specify the settings for the publishing action.
4. Run the job.
5. When it successfully completes, the specified results are published through the selected connection.

Create Connection

NOTE: Some connections require additional configuration outside of the application.

When a new connection is created, it is initially available only to you.

Prerequisites:

Before you create a new connection, please verify the following:

- On the datastore, you have read and (optionally) write locations.
- You have credentials to use to connect to this datastore. These credentials have permissions on your read /write locations.
- Some datastores require a special connection string, which must be inserted as part of the connection object.
- For more information, see *Create Connection Window*.

Read-only:

1. In the Import Data page, click the New icon in the left nav bar.
2. In the Create Connection window, specify the parameters of the connection.

Read-write:

1. In the Connections page, click Create Connection.
2. Click the connection category or search for a specific connection to create.
3. If a connection is grayed out:
 - a. It may already exist. Some connections types permit only one globally available connection.
 - b. It may not be supported in your product.
 - c. It may be read-only.
4. Click the name of the connection.
5. In the Create Connection window, specify the parameters of the connection.

Delete Connection

NOTE: You can delete a connection only if you are an admin or the connection owner, and the connection is not used to import any current datasets.

Steps:

1. In the Connections page, locate the connection to remove.
2. In the context menu, select **Delete....**
3. The connection is deleted.

Share a Connection

Contents:

- *Share a Connection*
- *Make a Connection Public*
- *Remove Sharing From a Connection*

This section provides an overview of sharing connections with other users for collaboration.

You can share connections with other users to use the same connection through the Connections page.

NOTE: Access to the Connections page in the application and privileges on connections is governed by roles in your workspace. For more information, please contact your workspace administrator.

Share a Connection

Steps:

1. From the Connections page, select the connection to share, and click **Share** from the context menu.
2. In the Share dialog, enter the name or email address of the user or users with whom to share the connection. You may be able to paste a comma-separated list of email addresses.
3. To provide view permissions or edit permissions, specify the privilege level of the user.
 - a. **Viewer:** User can access the connection and run jobs from it. User cannot modify the connection.
 - b. **Editor:** All of the viewer privileges, plus the user can edit the connection and share it with others.

Tip: Only an administrator can make a connection public.

4. You can specify whether to share your credentials with shared users:
 - a. **Share credentials:** (default) The credentials specified in the connection definition are shared with each user of the connection.
 - b. **Do not share credentials:** Each user must specify their own credentials.
5. To share the connection, click **Share**. The selected users can now see the connections in the **Shared with Me** tab of the Connections page.

Make a Connection Public

Only an administrator can make a connection public.

Remove Sharing From a Connection

You can remove the sharing from a connection by performing the following steps:

1. From the Share dialog for connections, select the user to remove sharing.
2. From the drop-down next to the user, Select **Remove**.
3. The sharing for the connection is removed.

Connection Types

Contents:

- *Configure*
 - *Disable Creating Connections for Non-Admins*
 - *Create Connection*
 - *Connection Categories*
 - *Applications*
 - *CRM ERP*
 - *Databases*
 - *File-API*
 - *Marketing*
 - *Connections to Base Storage Layer*
 - *Apache Hadoop HDFS - Cloudera*
 - *Amazon AWS*
 - *Microsoft Azure*
 - *Amazon S3 (layer)*
 - *Default Connections*
 - *Upload*
 - *Other Connections*
 - *Hive*
 - *Custom connections*
-

You can create the following types of connections from the Designer Cloud powered by Trifacta® platform . You can use the given links to create connection.

Notes:

- HDFS and Hive connections can be configured as part of platform configuration.
- Database connections should be configured after you have completed the platform configuration and have validated that it is working for locally uploaded files.

NOTE: Before creating connections to Hive or relational datastores, you must create and deploy an encryption key file. See *Create Encryption Key File*.

Configure

Disable Creating Connections for Non-Admins

By default, all users are permitted to create connections. As needed, you can disable the ability to create connections for non-admin users.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Search for the following parameter, and set it to `false`:

```
"webapp.connectivity.nonAdminManagementEnabled": true,
```


3. Save your changes and restart the platform.

Create Connection

1. Click **Connections** in the left nav bar.
2. In the Connections page, click **Create**.

Actions:

Search: Enter a search string in the search bar to locate connections of interest.

Tip: Search scans the selected category. For broadest search results, select the All types tab.

I'm interested: Click I'm interested to upvote adding the connection type to the Designer Cloud application . Help chart the future direction of connectivity!

For more information, see *Create Connection Window*.

Connection Categories

In the Create Connection window, connections are organized according to the following categories.

Applications

Connect to data storage for your web-based applications.

CRM ERP

These connections enable you to access data in your CRM or ERP systems.

Databases

These connections pertain to relational database sources.

NOTE: Unless otherwise noted, authentication to a relational connection requires basic authentication (username/password) credentials.

Enable: For more information, see *Relational Access*.

File-API

You can create connections to the listed file-based datastores or to other API-based storage.

Marketing

You can create connections to the datastores for popular marketing solutions.

Connections to Base Storage Layer

Connections of these type are automatically created for you.

Apache Hadoop HDFS - Cloudera

Enable: *Configure for Hadoop.*

Create New Connection: n/a

Amazon AWS

Running environment(s): Trifacta Photon and Spark

Base storage layer: S3

Microsoft Azure

Running environment(s): Trifacta Photon and Spark

Base storage layer: ADLS Gen1, ADLS Gen2, or WASB

For more information, see *Running Environment Options*.

For more information, see *Set Base Storage Layer*.

Amazon S3 (layer)

This connection type refers to using S3 as the base storage layer.

Supported Versions: n/a

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Not supported
Write	Not supported	Supported	Not supported

Enable: S3 Access

Create New Connection: n/a

Default Connections

These connections are automatically enabled and configured with the product.

Upload

Enable: Automatically enabled.

Create New Connection: n/a

Other Connections

Hive

Enable: *Configure for Hive*

NOTE: Additional configuration is required.

Create New Connection:

NOTE: A single public Hive connection is supported.

For more information, see *Hive Connections*.


Custom connections

For more information on other connectivity options, please contact *Alteryx Support*.

Oracle Database Connections

Contents:

- *Prerequisites*
 - *SSL*
- *Configure*
 - *Connection URL*
 - *Driver Information*
 - *Create via API*
- *Troubleshooting*
 - *"Could not create dataset - Lexical error"*
- *Use*
 - *SQL Syntax*
- *Data Conversion*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to one or more **Oracle Database** from the Designer Cloud powered by Trifacta® platform .

Tip: You can create connections to databases of this type that are managed by your enterprise or are hosted in cloud infrastructure. The required configuration is the same. The cloud-based version is labeled **on Amazon RDS** . In the Create Connection dialog, you can search for that term.

NOTE: You cannot disable type inference for Oracle sources. This is a known issue.

Supported Versions: 12.1.0.2

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Not supported
Write	Supported	Supported	Supported

Prerequisites

NOTE: Dots (.) in the names of Oracle tables or table columns are not supported.

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

SSL

If you are connecting to the Oracle Database using SSL, additional configuration is required in Oracle Database .

The Designer Cloud powered by Trifacta platform supports the use of the following SSL ciphers to communicate with Oracle Database :

```
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_WITH_RC4_128_MD5
SSL_RSA_WITH_DES_CBC_SHA
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA
SSL_DH_anon_WITH_RC4_128_MD5
SSL_DH_anon_WITH_DES_CBC_SHA
SSL_RSA_EXPORT_WITH_RC4_40_MD5
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
```

For more information, please see the documentation that is provided with your Oracle distribution.

Configure

To create this connection:

- In the Import Data page, click the Plus sign. Then, select the Relational tab. Click the Oracle Database card.
- You can also create connections through the Connections page. See *Connections Page*.

For additional details on creating an Oracle Database connection, see *Relational Access*. Modify the following properties as needed:

Property	Description
Host	Enter your hostname. Example: <div>testsql.database.windows.net</div>
Port	Set this value to 1521.
Connect String options	Please insert any connection options as a string here. See below.
Enable SSL	Select the option if the connection should use SSL. <div>NOTE: Additional configuration may be required in the database server. For more information, please consult the documentation that was provided with the distribution.</div> <div>NOTE: Additional configuration is required. See <i>Configure Data Service</i>.</div>
Service Name	Enter the name of the Oracle service.
User Name	(basic credential type only) Username to use to connect to the database.
Password	(basic credential type only) Password associated with the above username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Advanced options: Enable SSH Tunneling	This feature is not available.

Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connection URL

The properties that you provide are inserted into the following URL, which connects the Designer Cloud powered by Trifacta platform to the connection:

```
<host>:<port>/<service_name>
```

Connect string options

The connect string options are optional. If non-standard connections are required, Oracle Database supports using tnames format.

When the connect string options field is used:

- The connect string options parameters are prepended with `jdbc:oracle:thin:@`.
- The following fields are ignored from the form. These values must be specified as part of the tnames :
 - Host
 - Port
 - Service
 - SSL

After you specify the connect string options, the generated connection URL is automatically prepended with the following protocol information. Do not add this to the connection URL or connect string options:

```
jdbc:oracle:thin:@
```

Examples are below.

Use SID:

If you are using a service identifier, instead of a service name, please specify your connection string options as follows:

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=oracle.rds.example.com)(PORT=1521))(CONNECT_DATA=(SID=orcl)))
```

Use TCPS:

If TCPS protocol is required, you can specify your connection string options as follows:

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=oracle.rds.example.com)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=orcl)))
```

For more information, please see the documentation for the Oracle Database driver.

Driver Information

This connection uses the following driver:

- **Driver name:** `oracle.jdbc.driver.OracleDriver`
- **Driver version:** `com.oracle.database.jdbc:ojdbc8:19.9.0.0`
- **Driver documentation:** <https://docs.oracle.com/en/database/oracle/oracle-database/19/index.html>

Create via API

This connection can also be created using the API.

- Type: `jdbc`
- Vendor: `oracle`

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Troubleshooting

For more information on common error messages, see https://docs.oracle.com/cd/E11882_01/java.112/e16548/apxerrmsg.htm#JJDBC28962.

"Could not create dataset - Lexical error"

When you attempt to create a dataset with SQL from an Oracle database, you may receive an ORA error similar to the above. These queries may work in other database tools.

Solution:

The solution is to apply aliasing impacted columns in your SQL query. For more information, see *Supported SQL Syntax*.

Use

SQL Syntax

The following syntax requirements apply to this connection.

Object delimiter: double-quote

Example syntax:

Double quotes required around database and table names and not required around column names.

```
SELECT "column1","column2" FROM "databaseName"."tableName";
```

For more information on SQL in general, see *Supported SQL Syntax*.

Data Conversion

For more information on how values are converted during input and output with this database, see *Oracle Data Type Conversions*.

PostgreSQL Connections

Contents:

- *Prerequisites*
- *Configure*
 - *Connection URL*
 - *Driver Information*
 - *Create via API*
- *Troubleshooting*
- *Use*
 - *SQL Syntax*
- *Data Conversion*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to a PostgreSQL database from the Designer Cloud powered by Trifacta® platform . For more information on PostgreSQL , see <https://www.postgresql.org/>.

Tip: You can create connections to databases of this type that are managed by your enterprise or are hosted in cloud infrastructure. The required configuration is the same. The cloud-based version is labeled **on Amazon RDS** . In the Create Connection dialog, you can search for that term.

Supported Versions: 9.3.10

Supported Environments:

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Prerequisites

If the Trifacta databases are hosted on a PostgreSQL server, do not create a connection to this database.

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Configure

To create this connection:

- In the Import Data page, click the Plus sign. Then, select the Relational tab. Click the PostgreSQL card.
- You can also create connections through the Connections page. See *Connections Page*.

For additional details on creating a PostgreSQL connection, see *Relational Access*.

Modify the following properties as needed:

Property	Description
Host	Enter your fully qualified hostname. Example: <div>my.postgres.server</div>
Port	Set this value to 5432.
Connect String Options	Insert any additional connection parameters, if needed. See below.
Enable SSL	Select the checkbox to enable SSL connections to the database. <div>NOTE: Additional configuration may be required in the database server. For more information, please consult the documentation that was provided with the distribution.</div>
Database	Enter the name of the database on the server to which to connect.
User Name	Username to use to connect to the database.
Password	Password associated with the above username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Advanced options: Default Column Data Type Inference	Set to <code>disabled</code> to prevent the platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .
Advanced options: Enable SSH Tunneling	This feature is not available.
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connection URL

The properties that you provide are inserted into the following URL, which connects the Designer Cloud powered by Trifacta platform to the connection:

```
jdbc:postgresql://<host>:<port>/<database><connect-string-options>
```

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
?<prop1>=<val1>&<prop2>=<val2>...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

delimiters:

- ? : any set of connect string options must begin with a question mark.
- & : all additional property names must be prefixed with an ampersand (&).
- = : property names and values must be separated with an equal sign (=).

Driver Information

This connection uses the following driver:

- **Driver name:** `org.postgresql.Driver`
- **Driver version:** `org.postgresql:postgresql:42.1.1`
- **Driver documentation:** <https://jdbc.postgresql.org/documentation/head/index.html>

Create via API

This connection can also be created using the API.

- Type: `jdbc`
- Vendor: `postgres`

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Troubleshooting

Error message	Description
Class 08 Connection Exception	Connection failure: Designer Cloud application is unable to establish a connection.
Class 28 Invalid Authorization Specification	Typically, this error occurs when an invalid password has been submitted. Tip: Use the Test Connection button to validate your credentials.

For more information on error messages for this connection type, see <https://www.postgresql.org/docs/9.3/errcodes-appendix.html>.

NOTE: Please note the version number in the URL above.

Use

For more information, see *Database Browser*.

For more information on interacting with data, see *Using Databases*.

SQL Syntax

The following syntax requirements apply to this connection.

Object delimiter: double-quote

Example syntax:

Double quotes required around database, table names, and column names.

```
SELECT "column1", "column2" FROM "databaseName"."tableName";
```

For more information on SQL in general, see *Supported SQL Syntax*.


Data Conversion

For more information on how values are converted during input and output with this database, see *Postgres Data Type Conversions*.

MySQL Connections

Contents:

- *Prerequisites*
 - *Acquire MySQL Java driver*
- *Configure*
 - *Connection URL*
 - *Driver Information*
 - *Create via API*
- *Troubleshooting*
- *Use*
- *Data Conversion*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to one or more MySQL databases from the Designer Cloud powered by Trifacta® platform . For more information on MySQL, see <https://www.mysql.com/>.

Tip: You can create connections to databases of this type that are managed by your enterprise or are hosted in cloud infrastructure. The required configuration is the same. The cloud-based version is labeled **on Amazon RDS** . In the Create Connection dialog, you can search for that term.

Supported Versions: 5.7 and 8.0 Community

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Prerequisites

If the Trifacta databases are hosted on a MySQL server, do not create a connection to this database.

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Acquire MySQL Java driver

The MySQL Java driver is not packaged with the Designer Cloud powered by Trifacta platform installer. This driver must be downloaded and installed on the Trifacta node. The installation directory on the Trifacta node is the following:

```
/opt/trifacta/services/data-service/build/dependencies
```

For more information on acquiring the driver, see *Install Databases for MySQL*.

Configure

To create this connection:

- In the Import Data page, click the Plus sign. Then, select the Relational tab. Click the MySQL card.
- You can also create connections through the Connections page.
- See *Connections Page*.

For additional details on creating a MySQL connection, see *Relational Access*.

Modify the following properties as needed:

Property	Description
Host	Enter your fully qualified hostname. Example: <code>mysql-server.example.net</code>
Port	Set this value to 3306.
Connect String Options	Insert any additional connection parameters, if needed. See below.
User Name	Username to use to connect to the database.
Password	Password associated with the above username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Advanced options: Default Column Data Type Inference	Set to <code>disabled</code> to prevent the platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .
Advanced options: Enable SSH Tunneling	This feature is not available.
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connection URL

The properties that you provide are inserted into the following URL, which connects the Designer Cloud powered by Trifacta platform to the connection:

```
jdbc:mysql://<host>:<port>/<database><connect-string-options>
```

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
&<prop1>=<val1>&<prop2>=<val2>...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

delimiters:

- `&` : all additional property names must be prefixed with an ampersand (&).
- `=` : property names and values must be separated with an equal sign (=).

Default connect string options

The following connect string options are specified by default.

NOTE: These options should not be overridden or modified.

The following connect string option requires the driver to use cursor-based fetching to retrieve rows.

```
useCursorFetch=true;
```

Enable TLS (SSL)

You can insert the following connection string option to enable secure (TLS) connectivity with the MySQL server. Please note the TLS version numbers in the string listed below:

```
&enabledTLSProtocols=TLSv1,TLSv1.1,TLSv1.2
```

Set time zone

When data is imported from MySQL, the connector performs time zone adjustments to imported Datetime values.

Tip: By default, the MySQL driver adjusts the imported data to be represented in the time zone into which it is being read. If you are located in GMT -08:00, all data is rendered that your Trifacta user reads into the application through the MySQL connector is adjusted to this time zone.

These adjustments are performed in either of the following cases:

- The MySQL server is configured with a canonical time zone that is recognizable by Java (for example, Europe/Paris, Etc/GMT-5, UTC, etc.).
- The server's time zone is overridden by setting the Connector/J connection property `serverTimezone`.

In the latter case, you can specify the override time zone to apply as part of the connection string:

```
&serverTimezone=Europe/Paris
```

The following value sets the time zone of imported data to be UTC:

```
&serverTimezone=UTC
```

For more information on this behavior, see <https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-other-changes.html>.

Driver Information

This connection uses the following driver:

- **Driver name:** `com.mysql.cj.jdbc.Driver`
- **Driver version:** `8.0`
- **Driver documentation:** <https://dev.mysql.com/doc/connector-j/8.0/en/>

Create via API

This connection can also be created using the API.

- Type: `jdbc`
- Vendor: `mysql`

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Troubleshooting

Error message	Description
1042 - ER_BAD_HOST_ERROR	Unable to connect to host. Please verify the host and port values.
1045 - ER_ACCESS_DENIED_ERROR	Credentials failed to connect. Please verify your credentials. Tip: Click the Test Connection button to verify that your credentials are working properly.
Error: zero date value prohibited	Set the following option in the connect string options: <code>zeroDateTimeBehavior=convertToNull</code>
Prepared statement needs to be re-prepared.	Database Cursor is not compatible with PREPARED statements in MySQL. The fix is to set the following in the Connect String Options: <code>useCursorFetch=false</code>
SSLHandshakeException: No appropriate protocol (protocol is disabled or cipher suites are inappropriate)	SSL ciphers need to be enabled. For more information, see "Enable TLS (SSL)" above.

For more information on error messages for this connection type, see <https://dev.mysql.com/doc/refman/8.0/en/error-handling.html>.

Use

For more information, see *Database Browser*.


Data Conversion

For more information on how values are converted during input and output with this database, see *MySQL Data Type Conversions*.

Microsoft SQL Server Connections

Contents:

- *Prerequisites*
- *Configure*
 - *Connection URL*
 - *Authentication options*
 - *Driver Information*
 - *Create via API*
- *Troubleshooting*
- *Use*
 - *SQL Syntax*
- *Data Conversion*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to one or more Microsoft SQL Server databases from the Designer Cloud powered by Trifacta® platform .

Tip: You can create connections to databases of this type that are managed by your enterprise or are hosted in cloud infrastructure. The required configuration is the same. The cloud-based version is labeled **on Amazon RDS** . In the Create Connection dialog, you can search for that term.

Supported Versions: 12.0.4

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Prerequisites

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.
- If you plan to create an SSO connection of this type, additional configuration may be required. See *Enable SSO for Relational Connections*.

Configure

To create this connection:

- In the Import Data page, click the Plus sign. Then, select the Relational tab. Click the **Microsoft SQL Server** card.
- You can also create connections through the Connections page. See *Connections Page*.

For additional details on creating a Microsoft SQL Server connection, see *Relational Access*.

Modify the following properties as needed:

Property	Description
Host	Enter your hostname. Example: <div>testsql.database.windows.net</div>
Port	Set this value to 1433.
Connect String options	Insert any additional connection parameters, if needed. See below.
User Name	(basic credential type only) Username to use to connect to the database.
Password	(basic credential type only) Password associated with the above username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Advanced options: Default Column Data Type Inference	Set to <code>disabled</code> to prevent the platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .
Advanced options: Enable SSH Tunneling	This feature is not available.
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connection URL

The properties that you provide are inserted into the following URL, which connects Designer Cloud Powered by Trifacta Enterprise Edition to the connection:

```
jdbc:sqlserver://<host>:<port>;<prop1>=<val1>;<prop2>=<val2>
```

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
;<prop1>=<val1>;<prop2>=<val2>...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

delimiters:

- `;` : any set of connect string options must begin and end with a semi-colon.
- `=` : property names and values must be separated with an equal sign (=).

Example connect string options

The following connect string contains several options. Please insert as a single string (no line breaks):

```
;database=<database_name>;encrypt=true;trustServerCertificate=false;  
hostNameInCertificate=*.database.windows.net;loginTimeout=30;
```

Common connect string properties:

Property	Description
database	Set this value to <database_name>, the name of the database to which to connect.
encrypt	Set this value to true. Encrypted communication is required. <div>NOTE: You must deploy an encryption key file on the Trifacta node. For more information, see <i>Create Encryption Key File</i>.</div>
trustServerCertificate	When set to true, the Trifacta node does not validate the SQL Server TLS/SSL certificate. Default value is false.
hostNameInCertificate	Defines the host name in the server certificate. <div>NOTE: Do not modify this value unless required.</div>
loginTimeout	Number of seconds that the Trifacta node attempts to login to the database server. Default is 30.

Delimiters:

- ; : any set of connect string options must begin and end with a semi-colon.
- ; : all additional property names must be prefixed with a semi-colon.
- = : property names and values must be separated with an equal sign (=).

Authentication options

Kerberos:

You can use kerberos security for connecting to the database server. Additional configuration is required:

- For more information on enabling single-sign on for databases, see *Enable SSO for Relational Connections*.
- The SSO solution for databases leverages the platform's integration with cluster kerberos. For more information, *Configure for Kerberos Integration*.

Driver Information

This connection uses the following driver:

- **Driver name:** `com.microsoft.sqlserver.jdbc.SQLServerDriver`
- **Driver version:** `com.microsoft.sqlserver:mssql-jdbc:7.2.2.jre8`
- **Driver documentation:**
 - Overview:
<https://docs.microsoft.com/en-us/sql/connect/jdbc/overview-of-the-jdbc-driver?view=sql-server-ver15>
 - Connection URL:
<https://docs.microsoft.com/en-us/sql/connect/jdbc/building-the-connection-url?view=sql-server-ver15>
 - Connection properties:
<https://docs.microsoft.com/en-us/sql/connect/jdbc/setting-the-connection-properties?view=sql-server-ver15>

Create via API

This connection can also be created using the API.

- Type: jdbc
- Vendor: sqlserver

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Troubleshooting

Error message	Description
"The TCP/IP connection to the host <hostname>, port <port> has failed"	The host is not accessible.
"Login failed for user '<username>'."	Permission denied. Please verify your credentials. <div> Tip: Click the Test Connection button to check the connection credentials. </div>
"The certificate received from the remote server was issued by an untrusted certificate authority"	There was an issue with the trusted certificate on the SQL Server instance. To disable validation of the certificate, add the following to the connection string options: <pre>;trustServerCertificate=true;</pre>

Use

SQL Syntax

The following syntax requirements apply to this connection.

Object delimiter: none

Example syntax:

```
SELECT "column1","column2" FROM "databaseName"."tableName";
```

For more information on SQL in general, see *Supported SQL Syntax*.

For more information, see *Database Browser*.

Data Conversion

For more information on how values are converted during input and output with this database, see *SQL Server Data Type Conversions*.

Google Sheets Connections

Contents:

- *Limitations*
- *Enable*
- *Configure*
- *Test and Use*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Google Sheets provides a spreadsheet interface for cloud-based data stored in Google Drive. It allows multiple users to edit and format files in real-time. For more information, see <https://www.google.com/sheets/about/>.

You can create connections to your Google Sheets instance from the Designer Cloud powered by Trifacta® platform.

- You can create multiple Google Sheets connections.
- During connection creation, you must provide access to Google Drive.

Limitations

NOTE: This connection type is not supported for integration with a Hadoop-based running environment.

- This is a read-only connection.
- Single Sign-On (SSO) is not supported.
- Custom domains are not supported.
- Since data must be converted to a native file format, this connection does not support previewing of your data in the source.
- If you have enabled Google Advanced Protection, this connection type does not work.

Enable

- General relational connectivity must be enabled. For more information, see *Relational Access*.
- This connection type utilizes OAuth 2.0 for authentication. For more information, see *Enable OAuth 2.0 Authentication*.
- An OAuth 2.0 web client must be available for use in the Designer Cloud application. For more information, see *OAuth 2.0 for Google Sheets*.

Configure

To create this connection, select the **Google Sheets** card. For more information, see *Connections Page*.

Modify the following properties as needed:

Property	Description
OAuth 2.0 Client	The OAuth 2.0 Client is displayed. <div>NOTE: When you create the connection in this window, you must click Authenticate, which authenticates to the app. This step is required.</div>

Connection Name	Display name of the connection.
Connection Description	Description of the connection, which appears in the application.

Test and Use

Steps:

1. You can import datasets from Google Sheets through the Import Data page. For more information, see *Import Data Page*.
2. Perform a few simple transformations to the data. Run the job. For more information, see *Transformer Page*.
3. Verify the results.

External S3 Connections

Contents:

- *Prerequisites*
- *Permissions*
- *Limitations*
- *Create Connection*
 - *Create through application*
 - *Server Side KMS Key Identifier*
 - *Create via API*
 - *Java VFS Service*
- *Write*
 - *Uses of S3*
 - *Before you begin using S3*
 - *Secure access*
 - *Storing data in S3*
 - *Reading from sources in S3*
 - *Creating datasets*
 - *Writing results*
 - *Creating a new dataset from results*
 - *Purging files*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to specific S3 buckets through the Designer Cloud application . These connections to S3 enable workspace users to read from and write to specific S3 buckets.

Simple Storage Service (S3) is an online data storage service provided by Amazon, which provides low-latency access through web services. For more information, see <https://aws.amazon.com/s3/> .

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Not supported
Write	Supported	Supported	Not supported

Prerequisites

Before you begin, please verify that your Trifacta® environment meets the following requirements:

- **Integration:** Your Trifacta instance is connected to a running environment supported by your product edition.
- **Multiple region :** Multiple S3 connections can be configured in different regions.
- Verify that `Enable S3 Connectivity` has been enabled in the *Workspace Settings Page*.
- Acquire the Access Key ID and Secret Key for the S3 bucket or buckets to which you are connecting. For more information on acquiring your key/secret combination, contact your S3 administrator.

Permissions

Access to S3 requires:

- Each user must have appropriate permissions to access S3.

NOTE: If a user does not have write permissions to the specified S3 bucket, publishing jobs to the bucket fail.

- To browse multiple buckets through a single S3 connection, additional permissions are required. See below.

Limitations

- Authentication using IAM roles is not supported.
- Automatic region detection in the create and edit connection is not supported.
- Publishing the output to multi-part files is not supported.

NOTE: For some file formats, like Parquet, multi-part files are the default output.

- Publishing the output using compression option is not supported for Trifacta Photon jobs.

Workaround: If you need to generate an output using compression to this S3 bucket, you can run the job on another running environment.

Create Connection

You can create additional S3 connections by the following method:

Create through application

You can create a S3 connection through the application.

Steps:

1. Login to the application.
2. In the left navigation bar, click the **Connections** icon.
3. In the Create Connection page, click the External Amazon S3 card.
4. Specify the connection properties:

Property	Description
DefaultBucket	(Optional) The default S3 bucket to which to connect. When the connection is first accessed for browsing, the contents of this bucket are displayed. If this value is not provided, then the list of available buckets based on the key/secret combination is displayed when browsing through the connection. NOTE: To see the list of available buckets, the connecting user must have the getBucketList permission. If that permission is not present and no default bucket is listed, then the user cannot browse S3.
Access Key ID	Access Key ID for the S3 connection.
Secret Key	Secret Key for the S3 connection.

Server Side Encryption	If server-side encryption has been enabled on your bucket, you can select the server-side encryption policy to use when writing to the bucket. SSE-S3 and SSE-KMS methods are supported. For more information, see http://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html .
Server Side Kms key Id	When KMS encryption is enabled, you must specify the AWS KMS key ID to use for the server-side encryption. For more information, see "Server Side KMS Key Identifier" below.

For more information on the other options, see *Create Connection Window*.

5. Click **Save**.

Server Side KMS Key Identifier

When KMS encryption is enabled, you must specify the AWS KMS key ID to use for the server-side encryption.

- Access to the key:
 - Access must be provided to the authenticating user.
 - The AWS IAM role must be assigned to this key.
- Encrypt/Decrypt permissions for the specified KMS key ID:
 - Permissions must be assigned to the authenticating user.
 - The AWS IAM role must be given these permissions.
 - For more information, see <https://docs.aws.amazon.com/kms/latest/developerguide/key-policy-modifying.html>.

The format for referencing this key is the following:

```
"arn:aws:kms:<regionId>:<acctId>:key/<keyId>"
```

You can use an AWS alias in the following formats. The format of the AWS-managed alias is the following:

```
"alias/aws/s3"
```

The format for a custom alias is the following:

```
"alias/<FSR>"
```

where:

<FSR> is the name of the alias for the entire key.

Create via API

For more information on the vendor and type information to use, see *Connection Types*.

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

API: *API Reference*

- Type: remotefile
- vendor: aws

Java VFS Service

The Java VFS Service has been modified to handle an optional connection ID, enabling S3 URLs with connection ID and credentials. The other connection details are fetched through the Designer Cloud application to create the required URL and configuration.


```
// sample URI
s3://bucket-name/path/to/object?connectionId=136

// sample java-vfs-service CURL request with s3
curl -H 'x-trifacta-person-workspace-id: 1' -X GET 'http://localhost:41917/vfsList?uri=s3://bucket-name/path/to/object?connectionId=136'
```

Write

You can publish results to your external S3 buckets.

NOTE: The `append` action is not supported when publishing to S3.

Using S3 Connections

Uses of S3

The Designer Cloud powered by Trifacta platform can use S3 for the following tasks:

1. **Enabled S3 Integration:** The Designer Cloud powered by Trifacta platform has been configured to integrate with your S3 instance. For more information, see *S3 Access*.
2. **Creating Datasets from S3 Files:** You can read in source data stored in S3. An imported dataset may be a single S3 file or a folder of identically structured files. See *Reading from Sources in S3* below.
3. **Reading Datasets:** When creating a dataset, you can pull your data from a source in S3. See *Creating Datasets* below.
4. **Writing Results:** After a job has been executed, you can write the results back to S3. See *Writing Results* below.

In the Designer Cloud application, S3 is accessed through the S3 browser. See *S3 Browser*.

NOTE: When Designer Cloud application executes a job on a dataset, the source data is untouched. Results are written to a new location, so that no data is disturbed by the process.

Before you begin using S3

- **Access:** If you are using system-wide permissions, your administrator must configure access parameters for S3 locations. If you are using per-user permissions, this requirement does not apply. See *S3 Access*.

Avoid using `/trifacta/uploads` for reading and writing data. This directory is used by the Designer Cloud application.

- Your administrator should provide a writeable home output directory for you. This directory location is available through your user profile. See *Storage Config Page*.

Secure access

Your administrator can grant access on a per-user basis or for the entire workspace.

The Designer Cloud powered by Trifacta platform utilizes an S3 key and secret to access your S3 instance. These keys must enable read/write access to the appropriate directories in the S3 instance.

NOTE: If you disable or revoke your S3 access key, you must update the S3 keys for each user or for the entire system.

For more information, see [S3 Access](#).

Storing data in S3

Your administrator should provide raw data or locations and access for storing raw data within S3. All Trifacta users should have a clear understanding of the folder structure within S3 where each individual can read from and write results.

- Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.
- The Designer Cloud application stores the results of each job in a separate folder in S3.

NOTE: Designer Cloud Powered by Trifacta Enterprise Edition does not modify source data in S3. Source data stored in S3 is read without modification from source locations, and source data uploaded to the Designer Cloud powered by Trifacta platform is stored in `/trifacta/uploads`.

Reading from sources in S3

You can create an imported dataset from one or more files stored in S3.

NOTE: To be able to import datasets from the base storage layer, your user account must include the `dataAdmin` role.

NOTE: Import of glaciated objects is not supported.

Wildcards:

You can parameterize your input paths to import source files as part of the same imported dataset. For more information, see [Overview of Parameterization](#).

Folder selection:

When you select a folder in S3 to create your dataset, you select all files in the folder to be included.

Notes:

- This option selects all files in all sub-folders and bundles them into a single dataset. If your sub-folders contain separate datasets, you should be more specific in your folder selection.
- All files used in a single imported dataset must be of the same format and have the same structure. For example, you cannot mix and match CSV and JSON files if you are reading from a single directory.

When a folder is selected from S3, the following file types are ignored:

- `*_SUCCESS` and `*_FAILED` files, which may be present if the folder has been populated by the running environment.

NOTE: If you have a folder and file with the same name in S3, search only retrieves the file. You can still navigate to locate the folder.

Creating datasets

When creating a dataset, you can choose to read data in from a source stored from S3 or local file.

- S3 sources are not moved or changed.
- Local file sources are uploaded to `/trifacta/uploads` where they remain and are not changed.

Data may be individual files or all of the files in a folder. In the Import Data page, click the S3 tab. See *Import Data Page*.

Writing results

When you run a job, you can specify the S3 bucket and file path where the generated results are written. By default, the output is generated in your default bucket and default output home directory.

- Each set of results must be stored in a separate folder within your S3 output home directory.
- For more information on your output home directory, see *Storage Config Page*.

If Trifacta installation is using S3, do not use the `trifacta/uploads` directory. This directory is used for storing uploads and metadata, which may be used by multiple users. Manipulating files outside of the Designer Cloud application can destroy other users' data. Please use the tools provided through the Designer Cloud application interface for managing uploads from S3.

NOTE: When writing files to S3, you may encounter an issue where the UI indicates that the job failed, but the output file or files have been written to S3. This issue may be caused when S3 does not report the files back to the application before the S3 consistency timeout has expired. For more information on raising this timeout setting, see *S3 Access*.

Creating a new dataset from results

As part of writing results, you can choose to create a new dataset, so that you can chain together data wrangling tasks.

NOTE: When you create a new dataset as part of your results, the file or files are written to the designated output location for your user account. Depending on how your permissions are configured, this location may not be accessible to other users.

Purging files

Other than temporary files, the Designer Cloud powered by Trifacta platform does not remove any files that were generated or used by the platform, including:

- Uploaded datasets
- Generated samples
- Generated results

If you are concerned about data accumulation, you should create a bucket policy to periodically backup or purge directories in use. For more information, please see the S3 documentation.

Amazon Redshift Connections

Contents:

- *Prerequisites*
 - *Permissions*
- *Limitations*
- *Create Connection*
 - *Create through Designer Cloud application*
 - *Connection URL*
 - *Example*
 - *Driver Information*
 - *Create via API*
- *Troubleshooting*
- *Testing*
- *Using Redshift Connections*
 - *Uses of Redshift*
 - *Before you begin using Redshift*
 - *Secure access*
 - *Storing data in Redshift*
 - *Reading from Redshift*
 - *Writing to Redshift*
- *Reference*

This section provides information on how to enable Amazon Redshift connectivity and create one or more connections to Amazon Redshift sources.

- Amazon Redshift is a hosted data warehouse available through Amazon Web Services. It is frequently used for hosting of datasets used by downstream analytic tools such as Tableau and Qlik. For more information, see <https://aws.amazon.com/redshift/>.
- When exporting results, you can choose to write to a Redshift database. See *Publishing Dialog*.

Supported Environments:

NOTE: S3 must be set as the base storage layer. See *Set Base Storage Layer*.

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Not supported	Supported	Not supported
Write	Not supported	Supported	Not supported

Prerequisites

Before you begin, please verify that your Trifacta® environment meets the following requirements:

NOTE: If you are connecting to any relational source of data, such as Amazon Redshift or Oracle Database , you must add the Trifacta Service to your whitelist for those resources.

Tip: If the credentials used to connect to S3 do not provide access to Redshift, you can create an independent IAM role to provide access from Amazon Redshift to S3. If this separate role is available, the Amazon Redshift connection uses it instead. There may be security considerations.

1. **S3 base storage layer:** Amazon Redshift access requires use of S3 as the base storage layer, which must be enabled. See *Set Base Storage Layer*.
2. **Same region:** The Amazon Redshift cluster must be in the same region as the default S3 bucket.
3. **Integration:** Your Trifacta instance is connected to a running environment supported by your product edition.
4. **Deployment:** The Designer Cloud powered by Trifacta platform is deployed either on-premises or in EC2.

Permissions

Access to Amazon Redshift requires:

- Each user is able to access S3
- S3 is the base storage layer

If the credentials used to connect to S3 do not provide access to Amazon Redshift, you can create an independent IAM role to provide access from Amazon Redshift to S3. If this separate role is available, the Amazon Redshift connection uses it instead.

NOTE: There may be security considerations with using an independent role to govern this capability.

Steps:

1. The IAM role must contain the required S3 permissions. See *Required AWS Account Permissions*.
2. The Amazon Redshift cluster should be assigned this IAM role. For more information, see <https://docs.aws.amazon.com/redshift/latest/mgmt/authorizing-redshift-service.html>.

Limitations

- You can publish any specific job once to Amazon Redshift through the export window. See *Publishing Dialog*.
- When publishing to Redshift through the Publishing dialog, output must be in Avro or JSON format. This limitation does not apply to direct writing to Amazon Redshift.
- Management of nulls:
 - Nulls are displayed as expected in the Designer Cloud application.
 - When Amazon Redshift jobs are run, the UNLOAD SQL command in Redshift converts all nulls to empty strings. Null values appear as empty strings in generated results, which can be confusing. This is a known issue with Amazon Redshift.
- No schema validation is performed as part of writing results to Amazon Redshift.
- Credentials and permissions are not validated when you are modifying the destination for a publishing job.
- For Amazon Redshift, no validation is performed to determine if the target is a view and is therefore not a supported target.

Create Connection

You can create Amazon Redshift connections through the following methods.

Tip: SSL connections are recommended. Details are below.

Create through Designer Cloud application

Any user can create a Redshift connection through the application.

Steps:

1. Login to the application.
2. In the menu, click the Connections icon.
3. In the Create Connection page, click the **Amazon Redshift** connection card.
4. Specify the properties for your Amazon Redshift database connection:

Property	Description
Host	Hostname of the Amazon Redshift cluster NOTE: This value must be the full hostname of the cluster, which may include region information.
Port	Port number used to access the Amazon Redshift cluster. Default is 5439.
Connect String Options	Please insert any connection options as a string here. See below.
Database	The Amazon Redshift database to which to connect on the cluster
Credential Type	Options: Basic authentication with optional IAM role ARN: Basic authentication credentials specified in this window are used to connect to the Amazon Redshift database. Additional permissions may be governed by any ARN specified in the IAM role used for the account. Use this option if you are planning to specify a database username /password combination as part of the connection. IAM Role: Connection to Amazon Redshift is governed by the IAM role associated with the user's account.
Username	Username with which to connect to the Amazon Redshift database
Password	Password associated with the Amazon Redshift username
IAM Role ARN for Redshift /S3 connectivity	(Optional) You can specify an IAM role ARN that enables role-based connectivity between Amazon Redshift and the S3 bucket that is used as intermediate storage during Amazon Redshift bulk COPY/UNLOAD operations. Example: <code>arn:aws:iam::1234567890:role/MyRedshiftRole</code> For more information, see <i>Configure for EC2 Role-Based Authentication</i> .

For more information on the other options, see *Create Connection Window*.

5. Click **Save**.

Enable SSL connections

To enable SSL connections to Amazon Redshift, you must enable them first on your Amazon Redshift cluster. For more information, see <https://docs.aws.amazon.com/redshift/latest/mgmt/connecting-ssl-support.html>.

In your connection to Amazon Redshift, please add the following string to your Connect String Options:

```
;ssl=true
```

Save your changes.

Connection URL

The properties that you provide are inserted into the following URL, which connects Designer Cloud powered by Trifacta platform to the connection:

```
jdbc:redshift://<host>:<port>/<database><connect-string-options>
```

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
;<prop1>=<val1>&<prop2>=<val2>;...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

Delimiters:

- `;` : any set of connect string options must begin and end with a semi-colon.
- `;` : all additional property names must be prefixed with a semi-colon.
- `=` : property names and values must be separated with an equal sign (=).

Example

Access through AWS key-secret

The following example connection URL uses an AWS key/secret combination (IAM user) to access Amazon Redshift :

```
jdbc:redshift:iam://<redshift_clustername:region_name>:<port_number>/<database_name>?  
AccessKeyId=<access_key_value>&SecretAccessKey=<secret_key_value>&DBUser=<database_user_name>
```

where:

- `<redshift_clustername>`: the name of the Amazon Redshift cluster
- `<region_name>`: region identifier where the cluster is located
- `<port_number>`: port number to use to access the cluster
- `<database_name>`: name of the Redshift database to which to connect
- `<access_key_value>`: identifier for the AWS key
- `<secret_key_value>`: identifier for the AWS secret
- `<database_user_name>`: user identifier for connecting to the database

Access through IAM role and temporary credentials

The following example connection URL uses an AWS/Key secret combination using temporary credentials:

```
jdbc:redshift:iam://<redshift_clustername:region_name>:<port_number>/<database_name>?
AccessKeyId=<access_key_value>&SecretAccessKey=<secret_key_value>&SessionToken=<session_token>&DBUser=<database_user_name>
```

where:

- See previous.
- `<session_token>`: the AWS session token retrieved when using temporary credentials. The session token is requested by Designer Cloud powered by Trifacta platform when using AWS temporary credentials. For more information, see *Configure AWS Per-User Auth for Temporary Credentials*.

Driver Information

This connection uses the following driver:

- **Driver name:** `com.amazon.redshift.jdbc41.Driver`
- **Driver version:** `com.amazon.redshift:redshift-jdbc41-no-awssdk:1.2.45.1069`
- **Driver documentation:** <https://docs.aws.amazon.com/redshift/latest/mgmt/configure-jdbc-connection.html>

Create via API

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

API:

- Type: `redshift`
- vendor: `redshift`

Troubleshooting

For more information, see <https://docs.aws.amazon.com/redshift/latest/mgmt/troubleshooting-connections.html>.

Testing

Import a dataset from Amazon Redshift .

Add it to a flow, and specify a publishing action. Run a job.

NOTE: When publishing to Amazon Redshift through the Publishing dialog, output must be in Avro or JSON format. This limitation does not apply to direct writing to Amazon Redshift .

For more information, see *Verify Operations*.

After you have run your job, you can publish the results to Amazon Redshift through the Job Details page. See *Publishing Dialog*.

Using Redshift Connections

Uses of Redshift

Designer Cloud powered by Trifacta platform can use Redshift for the following tasks:

1. Create datasets by reading from Redshift tables.
2. Write to Redshift tables with your job results.
3. Ad-hoc publication of data to Redshift.

Before you begin using Redshift

- **Enable S3 Sources:** Redshift integration requires the following:
 - S3 is set to the base storage layer.
 - For more information, see [S3 Access](#).
- **Read Access:** Your Redshift administrator must configure read permissions. Your administrator should provide a database for upload to your Redshift datastore.
- **Write Access:** You can write and publish jobs results to Redshift.

Secure access

SSL is required.

Storing data in Redshift

Your Redshift administrator should provide database access for storing datasets. Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: Designer Cloud powered by Trifacta platform does not modify source data in Redshift. Datasets sourced from Redshift are read without modification from their source locations.

Reading from Redshift

You can create a Trifacta dataset from a table or view stored in Redshift.

NOTE: The Redshift cluster must be in the same region as the default S3 bucket.

NOTE: If a Redshift connection has an invalid iamRoleArn, you can browse and import datasets. However, any jobs executed using this connection fail. If the iamRoleArn is invalid, the only samples that you can generate are Quick Random samples; other sampling jobs fail.

For more information, see [Database Browser](#).

Writing to Redshift

NOTE: You cannot publish to a Redshift database that is empty. The database must contain at least one table.

You can write back data to Redshift using one of the following methods:

- Job results can be written directly to Redshift as part of the normal job execution.
- As needed, you can publish results to Redshift for previously executed jobs.

NOTE: You cannot re-publish results to Redshift if the original job published to Redshift. However, if the base job succeeded but publication to Redshift failed, you can publish from the Publishing dialog.

NOTE: To publish to Redshift, the source results must be in Avro or JSON format.

- For more information on how data is converted to Redshift, see [Redshift Data Type Conversions](#).

Data Validation issues:

- No validation is performed for the connection and any required permissions during job execution. So, you can be permitted to launch your job even if you do not have sufficient connectivity or permissions to access the data. The corresponding publish job fails at runtime.
- Prior to publication, no validation is performed on whether a target is a table or a view, so the job that was launched fails at runtime.

Reference

Supported Versions: n/a

Supported Environments:

NOTE: S3 must be set as the base storage layer. See *Set Base Storage Layer*.

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Not supported	Supported	Not supported
Write	Not supported	Supported	Not supported

AWS Glue Connections

Contents:

- *Prerequisites*
 - *Limitations*
 - *Create Connection*
 - *Create through application*
 - *Connection URL*
 - *Driver Information*
 - *Troubleshooting*
 - *Use*
 - *Testing*
 - *Using AWS Glue Connections*
 - *Enable*
 - *Uses of Glue*
 - *Before you begin using Glue*
 - *Secure Access*
 - *Reading partitioned data*
 - *Storing data in Glue*
 - *Reading from Glue*
 - *Notes on reading from views using custom SQL*
 - *Writing to Glue*
 - *SQL Syntax*
 - *Reference*
-

This section describes how to create a connection to your AWS Glue Data Catalog.

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Not supported
Write	Not supported	Not supported	Not supported

Prerequisites

Before you create a connection, you must enable *Designer Cloud Powered by Trifacta Enterprise Edition* to access AWS Glue. For more information, see *AWS Glue Access*.

Limitations

For more information, see "Supported Deployments" in *AWS Glue Access*.

Create Connection

You can create one or more connections to databases in your AWS Glue deployment.

Create through application

Any user can create an AWS Glue connection through the application.

Steps:

1. Login to the application.
2. In the menu, click **User menu > Preferences > Connections**.
3. In the Create Connection page, click the AWS Glue connection card.
4. Specify the properties for your AWS Glue connection. The following parameters are specific to AWS Glue connections:

Property	Description
EMR Master Node DNS	This DNS value can be retrieved from the EMR console.
Port	The port number through which to connect to the DNS master node
Connection String Options	No values are required here. Additional information is provided below.

For more information, see *Create Connection Window*.

Connection URL

The properties that you provide are inserted into the following URL, which connects Designer Cloud Powered by Trifacta Enterprise Edition to the connection:

```
jdbc:hive2://<host>:<port>/<database><connect-string-options>
```

where:

- `<database>` = name of the default database to which to connect. This value can be empty.

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
;<prop1>=<val1>;<prop2>=<val2>...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

Delimiters:

- `;` : any set of connect string options must begin with a semi-colon.
- `;` : sets of connect string options must separated by a semi-colon.
- `=` : property names and values must be separated with an equal sign (=).

Examples:

Designer Cloud Powered by Trifacta Enterprise Edition may insert additional authentication properties as part of the connect string options.

Driver Information

This connection uses the following driver:

- **Driver name:** `org.apache.hive.jdbc.HiveDriver`
- **Driver version:** The driver depends on the version of EMR that is in use.
- **Driver documentation:**
<https://cwiki.apache.org/confluence/display/Hive/HiveServer2+Clients#HiveServer2Clients-JDBC>

Troubleshooting

For more information, see <https://docs.aws.amazon.com/glue/latest/dg/troubleshooting-connection.html>.

Use

After the integration has been made between the platform and AWS Glue, you can import datasets.

- Import using custom SQL queries. For more information, see *Create Dataset with SQL*.

Testing

Import a dataset from AWS Glue. Add it to a flow, and run a job. Verify the results. For more information, see *Verify Operations*.

Using AWS Glue Connections

Enable

For more information, see *AWS Glue Access*.

Uses of Glue

The Designer Cloud powered by Trifacta platform can use Glue for the following tasks:

1. Create datasets by reading from Glue tables.

Before you begin using Glue

- **Read Access:** Your Glue administrator must configure read permissions to Glue databases.
- **Write Access:** Not supported.

Secure Access

For more information, see *Configure for AWS*.

Reading partitioned data

The Designer Cloud powered by Trifacta platform can read in partitioned tables. However, it cannot read individual partitions of partitioned tables.

Tip: If you are reading data from a partitioned table, one of your early recipe steps in the Transformer page should filter out the unneeded table data so that you are reading only the records of the individual partition.

Storing data in Glue

Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: The Designer Cloud powered by Trifacta platform does not modify source data in Glue. Datasets sourced from Glue are read without modification from their source locations.

Reading from Glue

You can create a Trifacta dataset from a table or view stored in Glue. For more information, see *Database Browser*.

Notes on reading from views using custom SQL

If you have enabled custom SQL and are reading data from a view, nested functions are written to a temporary filename, unless they are explicitly aliased.

Tip: If your custom SQL uses nested functions, you should create an explicit alias from the results. Otherwise, the job is likely to fail.

Problematic Example:

```
SELECT
  UPPER(`t1`.`column1`),
  TRIM(`t1`.`column2`),...
```

When these are read from a Glue view, the temporary column names are: `_c0`, `_c1`, etc. During job execution, Spark ignores the `column1` and `column2` reference.

Improved Example:

```
SELECT
  UPPER(`t1`.`column1`) as col1,
  TRIM(`t1`.`column2`) as col2,...
```

In this improved example, the two Glue view columns are aliased to the explicit column names, which are correctly interpreted and used by the Spark running environment during job execution.

Writing to Glue

Not supported.

SQL Syntax

The following syntax requirements apply to this connection.

Object delimiter: backtick

Example syntax:

```
SELECT `column1`,`column2` FROM `databaseName`.`tableName`;
```

For more information on SQL in general, see *Supported SQL Syntax*.

Reference

Supported Versions: n/a

Supported Environments:

NOTE: S3 must be set as the base storage layer, and the platform must be integrated with EMR. See *Set Base Storage Layer*.

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Not supported
Write	Not supported	Not supported	Not supported

Snowflake Connections

Contents:

- *Prerequisites*
 - *Prerequisites for OAuth 2.0*
 - *Limitations*
 - *Create Connection*
 - *Create through application*
 - *Connection URL*
 - *Driver Information*
 - *Create via API*
 - *Troubleshooting*
 - *Using Snowflake Connections*
 - *Uses of Snowflake*
 - *Before you begin using Snowflake*
 - *Secure access*
 - *Storing data in Snowflake*
 - *Reading from Snowflake*
 - *Using Snowflake as a running environment*
 - *Writing to Snowflake*
-

This section describes how to create a connection to your Snowflake datawarehouse.

- Snowflake is an S3-based data warehouse service hosted in the cloud. Auto-scaling, automatic failover, and other features simplify the deployment and management of your enterprise's data warehouse. For more information, see <https://www.snowflake.com>.

Tip: This connection type can be used to connect to Snowflake instances hosted in Azure.

Supported Environments:

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Not supported	Supported	Not supported
Write	Not supported	Supported	Not supported

Prerequisites

- **S3 base storage layer:** Snowflake access requires installation of Trifacta software in the AWS infrastructure and use of S3 as the base storage layer, which must be enabled. See *Set Base Storage Layer*.
- **Integration:** Your Trifacta instance is connected to an EMR cluster. See *Configure for EMR*.
- **Deployment:** Designer Cloud powered by Trifacta platform is deployed in EC2.

- Integration with Snowflake requires deployment of the Designer Cloud powered by Trifacta platform within a customer-managed AWS infrastructure. For more information, see *Snowflake Access*.
- **PUBLIC schema:** If you do not create an external staging database:
 - A `PUBLIC` schema is required in your default database.
 - If you do not provide a stage database, then a temporary stage is created for you under the `PUBLIC` schema in the default database.
- **S3 bucket:** The user-created stage must point to the same S3 bucket as the default bucket in use by the Designer Cloud powered by Trifacta platform .
- **Same region:** The Snowflake cluster must be in the same region as the default S3 bucket.
- **IAM role requirements:** If you are accessing AWS and Snowflake using IAM roles, please verify that the appropriate permissions have been assigned to the role to access Snowflake and its backing S3 buckets. For more information, see *Required AWS Account Permissions*.
- **Staging database:** Snowflake supports the use of a stage for reading and writing data to S3 during job executions.

NOTE: If a stage is not deployed, then the user must have write permissions to the default database, which is used instead for staging your data in Snowflake . These permissions must be included in the AWS credentials applied to the user account.

For more information, see *Snowflake Access*.

Prerequisites for OAuth 2.0

If you are connecting to your Snowflake deployment using OAuth 2.0 authentication, additional configuration is required:

- OAuth 2.0 must be enabled and configured for use in the product. For more information, see *Enable OAuth 2.0 Authentication*.
- OAuth 2.0 requirements:
 - Create a security integration in your Snowflake deployment.
 - Create an OAuth 2.0 client in the Designer Cloud application that connects using the security integration.
 - For more information, see *OAuth 2.0 for Snowflake*.

Limitations

- You cannot perform ad-hoc publication to Snowflake .
- SSO connections are not supported.
- To ingest data from a Snowflake table, one of the following must be enabled:
 - A named stage must be created for the table. For more information, see the Snowflake documentation.
 - Snowflake must be permitted to create a temporary stage, which requires:
 - Write permissions on the table's database, and
 - A schema named `PUBLIC` must exist and be accessible.
- No schema validation is performed as part of writing results to Snowflake .
- Credentials and permissions are not validated when you are modifying the destination for a publishing job.
- For Snowflake , no validation is performed to determine if the target is a view and is therefore not a supported target.

Create Connection

You can create Snowflake connections through the following methods.

Create through application

Any user can create a Snowflake connection through the application.

Steps:

1. Login to the application.
2. In the left nav bar, click the Connections icon.
3. In the Create Connection page, click the Snowflake connection card.
4. Specify the properties for your Snowflake database connection. The following parameters are specific to Snowflake connections:

NOTE: In Snowflake connections, property values are case-sensitive. Snowflake -related locations are typically specified in capital letters.

Property	Description
Account Name	<p>Snowflake account to use. Suppose your hostname is the following:</p> <div>mycompany.snowflakecomputing.com</div> <p>Your account name is the following:</p> <div>mycompany</div> <p>NOTE: Your full account name might include additional segments that identify the region and cloud platform where your account is hosted.</p>
Warehouse	<p>The name of the warehouse to use when connected. This value can be an empty string.</p> <p>If specified, the warehouse should be an existing warehouse for which the default role has privileges.</p>
Stage	<p>If you have deployed a Snowflake stage for managing file conversion to tables, you can enter its name here. A stage is a database object that points to an external location on S3. It must be an external stage containing access credentials.</p> <p>If a stage is used, then this value is typically the schema and the name of the stage. Example value:</p> <div>MY_SCHEMA.MY_STAGE</div> <p>If a stage is not specified, a temporary stage is created using the current user's AWS credentials.</p> <p>NOTE: Without a defined stage, you must have write permissions to the database from which you import. This database is used to create the temporary stage.</p> <p>For more information on stages, see https://docs.snowflake.net/manuals/sql-reference/sql/create-stage.html.</p>
Credential Type	<p>Select the type of credentials to provide with the connection:</p> <ul style="list-style-type: none">• Basic - username and password are used by the connection to authenticate to Snowflake .

	<ul style="list-style-type: none"> OAuth 2.0 - use OAuth 2.0 client connect to Snowflake . The client must already be defined in the Designer Cloud application and then selected in the connection configuration. <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>NOTE: After you have specified the connection to use OAuth 2.0, click Authenticate to validate the connection with the target datastore. If you have modified the connection, click Re-authenticate to validate the new connection definition. You must re-authenticate if you receive an expired tokens message. For more information, see <i>Enable OAuth 2.0 Authentication</i>.</p> </div> <p>For more information, see <i>OAuth 2.0 for Snowflake</i>.</p>
Database for Stage	<p>(optional) If you are using a Snowflake stage, you can specify a database other than the default one to host the stage.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>NOTE: If you are creating a read-only connection to Snowflake , this field is required. The accessing user must have write permission to the specified database.</p> </div> <p>If no value is specified, then your stage must be in the default database.</p>

For more information, see *Create Connection Window*.

Connection URL

The properties that you provide are inserted into the following URL, which connects the Designer Cloud powered by Trifacta platform to the connection:

```
jdbc:snowflake://<account_name>.snowflakecomputing.com/?db=<database>&warehouse=<warehouse><connect-string-options>
```

where:

- <database> = name of the default database to which to connect. This value can be empty.

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
&<prop1>=<val1>&<prop2>=<val2>...
```

where:

- <prop> : the name of the property
- <val> : the value for the property

Delimiters:

- & : any set of connect string options must begin with an ampersand (&).
- = : property names and values must be separated with an equal sign (=).

Disable SSL connections

By default, connections to Snowflake use SSL. To disable, please add the following string to your Connect String Options:

```
;ssl=false
```

Connect through proxy

If you require connection to Snowflake through a proxy server, additional Connect String Options are required. For more information, see <https://docs.snowflake.net/manuals/user-guide/jdbc-configure.html#specifying-a-proxy-server-in-the-jdbc-connection-string>

Driver Information

This connection uses the following driver:

- **Driver name:** `net.snowflake.client.jdbc.SnowflakeDriver`
- **Driver version:** `net.snowflake:snowflake-jdbc:3.8.5`
- **Driver documentation:** <https://docs.snowflake.com/en/user-guide/jdbc.html>

Create via API

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnectionAPI>:

- Type: `snowflake`
- vendor: `snowflake`

Troubleshooting

Error Message	Description
Null values in some columns for all rows	When there are spaces/special characters in columns names, null values can be inserted for all rows in the column. The workaround is to remove any special characters and spaces from column names.

Using Snowflake Connections

Uses of Snowflake

The Designer Cloud powered by Trifacta platform can use Snowflake for the following tasks:

1. Create datasets by reading from Snowflake tables.
2. Write to Snowflake tables with your job results.

Before you begin using Snowflake

- **Enable S3 Sources: Snowflake** integration requires the following:
 - Installation of the product on a customer-managed AWS infrastructure.
 - S3 is set to the base storage layer.
 - For more information, see *Snowflake Access*.
- **Read Access:** Your Snowflake administrator must configure read permissions. Your administrator should provide a database for upload to your Snowflake data warehouse.
 - **Read-only Access:** If you are creating a read-only connection to Snowflake, you must provide a database for staging. The accessing user must have write permission to the specified database.
- **Write Access:** You can write and publish jobs results to Snowflake.

Secure access

SSL is the default connection method.

Storing data in Snowflake

Your Snowflake administrator should provide database access for storing datasets. Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: The **Designer Cloud powered by Trifacta platform** does not modify source data in Snowflake . Datasets sourced from Snowflake are read without modification from their source locations.

Reading from Snowflake

You can create a Trifacta dataset from a table stored in Snowflake .

NOTE: The Snowflake cluster must be in the same region as the default S3 bucket.

Using Snowflake as a running environment

If your source data and output targets are located in Snowflake , you may be able to execute your transformations inside Snowflake .

Writing to Snowflake

You can write back data to Snowflake using one of the following methods:

- Job results can be written directly to Snowflake as part of the normal job execution. Create a new publishing action to write to Snowflake .
- For more information on how data is converted to Snowflake , see *Snowflake Data Type Conversions*.

Data Validation issues:

- No validation is performed for the connection and any required permissions during job execution. So, you can be permitted to launch your job even if you do not have sufficient connectivity or permissions to access the data. The corresponding publish job fails at runtime.
- Prior to publication, no validation is performed on whether a target is a table or a view, so the job that was launched fails at runtime.

Azure SQL Database Connections

Contents:

- Limitations
- Prerequisites
- Configure
 - Configure for SSO
- Use
- Data Conversion

You can create a connection to a Microsoft Azure SQL Database from Designer Cloud Powered by Trifacta® Enterprise Edition. This section describes how to create connections of this type.

- This connection type supports data ingestion into ADLS/WASB. When large volumes of data are read from an Azure SQL Database during job execution, the data is stored in a temporary location in ADLS/WASB. After the job has been executed, the data is removed from the datastore. This process is transparent to the user.
- For more information on Azure SQL Database , see <https://azure.microsoft.com/en-us/services/sql-database/>.

NOTE: This database connection is a specialized version of a SQL Server connection.

NOTE: For Azure deployments, some additional configuration properties must be applied. See *Configure for Azure* .

Supported Versions: Azure SQL Database version 12 (other versions are not supported)

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Not supported	Supported
Write	Supported	Not supported	Supported

Limitations

- Connections of this type cannot be created via API.

Prerequisites

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Configure

- For additional details on creating an Azure SQL Database connection, see *Relational Access*.

Please create an Azure SQL Database connection and then specify the following properties with the listed values:

Property	Description
----------	-------------

Host	<p>Enter your hostname. Example:</p> <pre>testsql.database.windows.net</pre>
Port	Set this value to 1433.
Connect String options	<p>Please insert the following as a single string (no line breaks):</p> <pre>;encrypt=true;trustServerCertificate=false; hostNameInCertificate=*.database.windows.net;loginTimeout=30;</pre> <div> <p>Tip: If you have access to the Azure SQL Database through the Azure SQL Database Portal, please copy the Connect String from that configuration. You may omit the username and password from that version of the string.</p> </div>
Database	(optional) Name of the Azure SQL Database to which you are connecting.
User Name	(for basic Credential Type) Username to use to connect to the database.
Password	(for basic Credential Type) Password associated with the above username.
Credential Type	<ul style="list-style-type: none"> • <code>basic</code> - Specify username and password as part of the connection • <code>Azure Token SSO</code> - Use the SSO principal of the user creating the connection to authenticate to the Azure SQL Database . Additional configuration is required. See <i>Enable SSO for Azure Relational Connections</i>.
Default Column Data Type Inference	Set to <code>disabled</code> to prevent Designer Cloud Powered by Trifacta Enterprise Edition from applying its own type inference to each column on import. The default value is <code>enabled</code> .

Configure for SSO

If you have enabled Azure AD SSO integration for the Designer Cloud powered by Trifacta platform , you can create SSO connections to Azure relational databases. See *Enable SSO for Azure Relational Connections*.

Use

For more information, see *Database Browser*.

Data Conversion

For more information on how values are converted during input and output with this database, see *SQL Server Data Type Conversions*.

Microsoft SQL Data Warehouse Connections

Contents:

- Overview
 - Table types
 - SQL pool types
- Limitations
- Prerequisites
- Azure Synapse Analytics (Formerly Microsoft SQL DW) permissions
- Azure Synapse Analytics (Formerly Microsoft SQL DW) External Data Source Name
- Configure
 - Configure for SSO
- Use
- Data Conversion

This section describes how to create connections to Azure® Synapse Analytics (Formerly Microsoft® SQL DW)® .

Tip: This connection is now known as Azure Synapse Analytics.

Supported Versions: n/a

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Not supported	Not supported	Supported
Write	Not supported	Not supported	Supported

Overview

Table types

Azure Synapse Analytics (Formerly Microsoft SQL DW) can interact with the following table types:

Table type	Description
Managed table	Managed tables are database tables that are specifically defined within the database server. Read and write are supported.
External table	External tables are references to files on the backend storage layer on top of which is a database schema. The table is defined by reading and writing through the database schema to the underlying file storage. <div>NOTE: When publishing to an external table, the output file type is Parquet.</div> <div>NOTE: When publishing to an external table under ADLS user mode, the system credentials are used to write to the storage location and must have the appropriate permissions.</div>

SQL pool types

Azure Synapse Analytics (Formerly Microsoft SQL DW) supports two different SQL pooling methods through which you connect to your data managed by the database server.

Tip: The type of SQL pooling in use is determined by the URL that you use to connect to Azure Synapse Analytics (Formerly Microsoft SQL DW) . URLs with `ondemand` in them are for serverless SQL pool connections.

Dedicated SQL pool

These connections utilize a fixed and dedicated set of SQL pool resources. The admin user can define the size and availability of these resources for performing work.

This connection requires more permissions. You must also specify an External Datasource Name. See below.

Tip: Spark-based jobs that read or write through your Azure Synapse Analytics (Formerly Microsoft SQL DW) connection leverage PolyBase for faster performance.

Supported table types: Managed tables, external tables

Serverless SQL pool

These connections specify the SQL pool resources based on the size of the job. In theory, these connection types can scale infinitely for jobs of any size.

Tip: This connection requires fewer permissions on the data warehouse and its databases but is less performant. The URL for these connections always contain `ondemand`.

Supported table types: External tables only

Limitations

- Azure Synapse Analytics (Formerly Microsoft SQL DW) connections are available only if you have deployed the Designer Cloud powered by Trifacta® platform onto Azure.
- SSL connections to Azure Synapse Analytics (Formerly Microsoft SQL DW) are required.

NOTE: In this release, this connection cannot be created through the APIs. Please create connections of this type through the application.

NOTE: Under Azure SSO, write operations are not supported through Azure Synapse Analytics (Formerly Microsoft SQL DW) connections.

- JSON files cannot be read in through this connection type.
- For custom SQL and file formats other than CSV and Parquet, data is read through CETAS (create external table and select).
 - In some cases, reading from CETAS tables may exceed 30 minutes, which is the read limit imposed by Azure. These jobs time out.
 - In timeout situations, you may be able to fall back to a direct JDBC read of these sources.
- When publishing to an external table, the output file type is always Parquet.

Prerequisites

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Azure Synapse Analytics (Formerly Microsoft SQL DW) permissions

- **Read:** The authenticating DB user must have read permissions to any Azure Synapse Analytics (Formerly Microsoft SQL DW) , schemas and tables to which the user should have access.
- **Write:** In addition to the above, the authenticating DB user must have the following permissions:

```
CREATE TABLE**
ALTER ANY SCHEMA
ALTER ANY EXTERNAL DATA SOURCE
ALTER ANY EXTERNAL FILE FORMAT
```

- The authenticating DB user must also have read access to the external data source.

Azure Synapse Analytics (Formerly Microsoft SQL DW) External Data Source Name

When specifying a connection to external tables, you can provide an External Data Source Name value as part of the connection definition. The External Data Source enables publishing and support for large-scale data ingestion.

When the External Data Source is provided:

- CETAS (create external table and select) is used for reading in data.

If the External Data Source is not provided:

- JDBC read is used for reading in data.
- The connection is read-only.
- The connection must be to a set of managed tables.
- The native ingestion of the Designer Cloud powered by Trifacta platform is used.

Requirements:

- The external data source must be created by the database admin on the default database defined in the connection. For more information, see <https://docs.microsoft.com/en-us/sql/t-sql/statements/create-external-data-source-transact-sql?view=azure-sqldw-latest&tabs=dedicated>
- The External Data Source must point to the same storage location as the base storage layer for the Designer Cloud powered by Trifacta platform . For example, if the base storage layer is WASB, the External Datasource must point to the same storage account defined in Trifacta configuration. If this configuration is incorrect, then publishing and ingestion of data fail.
- For more information on privileges required for the authenticating DB user, see <https://docs.microsoft.com/en-us/sql/t-sql/statements/create-external-table-transact-sql>.

Configure

To create this connection, see *Connections Page*.

For additional details on creating a relational connection, see *Relational Access*.

Please create a connection of this type and modify the following properties with the listed values:

--	--

Property	Description
Host	<p>Enter your hostname. Example:</p> <div>testsql.database.windows.net</div> <div> Tip: If your Host value contains <code>ondemand</code>, then you are using serverless SQL pools. </div>
Port	Set this value to 1433.
Database	Set this value to the default database name.
External Data Source Name	For external table connections, you must provide an External Data Source. See above for details.
Connect String options	Include any options required for your environment:
User Name	Username to use to connect to the database.
Password	Password associated with the above username.
Credential Type	<ul style="list-style-type: none"> <code>basic</code> - Specify username and password as part of the connection <code>Azure Token SSO</code> - Use the SSO principal of the user creating the connection to authenticate to the SQL Server database. Additional configuration is required. See <i>Enable SSO for Azure Relational Connections</i>.
Default Column Data Type Inference	Set to <code>disabled</code> to prevent the Designer Cloud powered by Trifacta platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .

Configure for SSO

If you have enabled Azure AD SSO integration for the Designer Cloud powered by Trifacta platform , you can create SSO connections to Azure relational databases.

NOTE: When Azure AD SSO is enabled, write operations to Azure Synapse Analytics (Formerly Microsoft SQL DW) are not supported.

See *Enable SSO for Azure Relational Connections*.

Use

For more information on locating data, see *Database Browser*.

For more information, see *Using SQL DW*.

For more information on defining output objects, see *Microsoft SQL Data Warehouse Table Settings*.


Data Conversion

For more information on how values are converted during input and output with this database, see *SQL DW Data Type Conversions*.

AlloyDB Connections

Contents:

- *Prerequisites*
- *Configure*
 - *Connection URL*
 - *Driver Information*
 - *Create via API*
- *Troubleshooting*
- *Use*
 - *SQL Syntax*
- *Data Conversion*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to a Alloy DB database from Designer Cloud Powered by Trifacta® Enterprise Edition. For more information on Alloy DB , see <https://www.postgresql.org/>.

Tip: You can create connections to databases of this type that are managed by your enterprise or are hosted in cloud infrastructure. The required configuration is the same. The cloud-based version is labeled **on Amazon RDS** . In the Create Connection dialog, you can search for that term.

Supported Versions: 9.3.10

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Prerequisites

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Configure

To create this connection:

- In the Import Data page, click the Plus sign. Then, select the Relational tab. Click the Alloy DB card.
- You can also create connections through the Connections page. See *Connections Page*.

For additional details on creating a Alloy DB connection, see *Relational Access*.

Modify the following properties as needed:

Property	Description
----------	-------------

Host	Enter your fully qualified hostname. Example: <code>my.postgres.server</code>
Port	Set this value to 5432.
Connect String Options	Insert any additional connection parameters, if needed. See below.
Enable SSL	Select the checkbox to enable SSL connections to the database. NOTE: Additional configuration may be required in the database server. For more information, please consult the documentation that was provided with the distribution.
Database	Enter the name of the database on the server to which to connect.
User Name	Username to use to connect to the database.
Password	Password associated with the above username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Advanced options: Default Column Data Type Inference	Set to <code>disabled</code> to prevent the platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .
Advanced options: Enable SSH Tunneling	This feature is not available.
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connection URL

The properties that you provide are inserted into the following URL, which connects Designer Cloud Powered by Trifacta Enterprise Edition to the connection:

```
jdbc:postgresql://<host>:<port>/<database><connect-string-options>
```

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
?<prop1>=<val1>&<prop2>=<val2>...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

delimiters:

- `?` : any set of connect string options must begin with a question mark.
- `&` : all additional property names must be prefixed with an ampersand (&).
- `=` : property names and values must be separated with an equal sign (=).

Driver Information

This connection uses the following driver:

- **Driver name:** `org.postgresql.Driver`
- **Driver version:** `org.postgresql:postgresql:42.1.1`
- **Driver documentation:** <https://jdbc.postgresql.org/documentation/head/index.html>

Create via API

This connection can also be created using the API.

- Type: `jdbc`
- Vendor: `postgres`

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Troubleshooting

Error message	Description
Class 08 Connection Exception	Connection failure: the web client or Trifacta node is unable to establish a connection.
Class 28 Invalid Authorization Specification	Typically, this error occurs when an invalid password has been submitted. <div>Tip: Use the Test Connection button to validate your credentials.</div>

For more information on error messages for this connection type, see <https://www.postgresql.org/docs/9.3/errcodes-appendix.html>.

NOTE: Please note the version number in the URL above.

Use

For more information, see *Database Browser*.

For more information on interacting with data, see *Using Databases*.

SQL Syntax

The following syntax requirements apply to this connection.

Object delimiter: double-quote

Example syntax:

Double quotes required around database, table names, and column names.

```
SELECT "column1", "column2" FROM "databaseName"."tableName";
```

For more information on SQL in general, see *Supported SQL Syntax*.

Data Conversion

For more information on how values are converted during input and output with this database, see *Postgres Data Type Conversions*.

Databricks Tables Connections

Contents:

- *Limitations*
 - *Prerequisites*
 - *Insert Databricks Access Token*
 - *Enable*
 - *Create Connection*
 - *Connection URL*
 - *Driver Information*
 - *Data Conversion*
 - *Create via API*
 - *Troubleshooting*
 - *Failure when importing wide Databricks Tables table*
 - *Using Databricks Table Connections*
 - *Uses of Databricks tables*
 - *Before you begin using Databricks tables*
 - *Storing data in Databricks tables*
 - *Reading from Databricks Tables*
 - *Writing to Databricks Tables*
 - *Ad-hoc Publishing to Databricks Tables*
 - *Reference*
-

You can create a connection to Databricks Tables from the Designer Cloud powered by Trifacta platform .

This section describes how to create connections of this type.

- Databricks Tables provides a JDBC-based interface for reading and writing datasets in ADLS or WASB. Using the underlying JDBC connection, you can access your ADLS or WASB data like a relational datastore, run jobs against it, and write results back to the datastore as JDBC tables.
- Your connection to Databricks Tables leverages the SSO authentication that is native to Databricks .
 - For more information on Azure Databricks Tables, see <https://docs.microsoft.com/en-us/azure/databricks/data/tables>.

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Not supported	Supported	Supported
Write	Not supported	Supported	Supported

Limitations

- Ad-hoc publishing of generated results to Databricks Tables is not supported.
- Integration with Kerberos or secure impersonation is not supported.
- Some table types and publishing actions are not supported.
- Access to external Hive metastores is not supported.

Prerequisites

- **Azure:** The Designer Cloud powered by Trifacta platform must be installed on Azure and integrated with an Azure Databricks cluster.
 - See *Install for Azure*.

- See *Configure for Azure Databricks*.

NOTE: For job execution on Spark, the connection must use the Spark instance on the Azure Databricks cluster. No other Spark instance is supported. You can run jobs from this connection through the Photon running environment. For more information, see *Running Environment Options*.

- **AWS:** The Designer Cloud powered by Trifacta platform must be installed on AWS and integrated with an AWS Databricks cluster.
 - See *Install for AWS*.
 - See *Configure for AWS Databricks*.

NOTE: For job execution on Spark, the connection must use the Spark instance on the AWS Databricks cluster. No other Spark instance is supported. You can run jobs from this connection through the Photon running environment. For more information, see *Running Environment Options*.

- This connection interacts with Databricks Tables through the Hive metastore that has been installed in the Databricks cluster.

NOTE: External Hive metastores are not supported.

Insert Databricks Access Token

Each user must insert a Databricks Personal Access Token into the user profile. For more information, see *Databricks Settings Page*.

Enable

To enable Databricks Tables connections, please complete the following:

NOTE: Typically, you need only one connection to Databricks Tables, although you can create multiple connections.

NOTE: This connection is created with SSL automatically enabled.

Tip: Your SSL certificate should be created using 2047-bit or larger keys. In some environments, such as RHEL 8.x, 1024-bit encryption is no longer accepted per default cryptographic policies.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameter and set it to `true`:

```
"feature.databricks.connection.enabled": true,
```

3. To allow for direct publishing of job results to Databricks tables from the Run Job page, you must enable the following parameters. For more information on these settings, see *Databricks Tables Table Settings*.

Parameter	Description
feature.databricks.enableDeltaTableWrites	Set this value to <code>true</code> to enable users to choose to write generated results to Databricks delta tables from the Run Job page.
feature.databricks.enableExternalTableWrites	Set this value to <code>true</code> to enable users to choose to write generated results to Databricks external tables from the Run Job page.

4. Save your changes and restart the platform.

Create Connection

This connection can also be created via API. For details on values to use when creating via API, see *Connection Types*.

Please create a Databricks connection and then specify the following properties with the listed values:

NOTE: Host and port number connection information is taken from Databricks and does not need to be re-entered here.

- See *Configure for Azure Databricks*.
- See *Configure for AWS Databricks*.

Property	Description
Connect String options	Please insert any connection string options that you need. Connect String options are not required for this connection.
Test Connection	Click this button to test the specified connection.
Default Column Data Type Inference	Set to <code>disabled</code> to prevent the Designer Cloud powered by Trifacta platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .

Connection URL

The properties that you provide are inserted into the following URL, which connects Designer Cloud Powered by Trifacta Enterprise Edition to the connection:

```
jdbc:spark://<host>:<port>/<database><connect-string-options>
```

The Connection URL is mostly built up automatically using cluster configuration for the platform.

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
;<prop1>=<val1>;<prop2>=<val2>...
```

where:

- `<prop>` : the name of the property
- `<val>` : the value for the property

delimiters:

- `;` : any set of connect string options must begin and end with a semi-colon.
 - A semi-colon can be omitted from the end of the connect string options.
- `=` : property names and values must be separated with an equal sign (=).

Use HTTP

To enable the use of the HTTP protocol, specify the following in the connect string options:

```
;transportMode=http;
```

Use SSL

To enable the use of SSL for the connection, specify the following in the connect string options:

```
;ssl=1;
```

HTTP Path

When HTTP is enabled, you can specify the path as a connect string option:

```
;httpPath=sql/protocolv1/o/0/xxxx-xxxxxx-xxxxxxxx;
```

Authentication

You can specify a Databricks personal access token to use when authenticating to the database using the following connect string options.

```
;AuthMech=3;UID=token;PWD=<Databricks-personal-access-token>
```

where:

- `<Databricks-personal-access-token>` = the personal access token of the user who is connecting to the database.

Driver Information

This connection uses the following driver:

- **Driver name:** `com.simba.spark.jdbc41.Driver`
- **Driver version:** `com.simba.jdbc:SparkJDBC41:2.6.11.1014`
- **Driver documentation:** <https://docs.databricks.com/integrations/bi/jdbc-odbc-bi.html>

Data Conversion

For more information on how values are converted during input and output with this database, see *Databricks Tables Data Type Conversions*.

Create via API

API: API Reference

- Type: jdbc
- Vendor: databricks

Troubleshooting

For more information on error messages for this connection type, see <https://kb.databricks.com/bi/jdbc-odbc-troubleshooting.html>.

Failure when importing wide Databricks Tables table

If you are attempting to import a table containing a large number of columns (>200), you may encounter an error message similar to the following:

```
org.apache.spark.SparkException: Job aborted due to stage failure: Task 0 in stage 408.0 failed 4 times, most recent failure: Lost task 0.3 in stage 408.0 (TID 1342, 10.139.64.11, executor 11): org.apache.spark.SparkException: Kryo serialization failed: Buffer overflow. Available: 0, required: 1426050. To avoid this, increase spark.kryoserializer.buffer.max value.
```

The problem is that the serializer ran out of memory.

Solution:

To address this issue, you can increase the Kryoserializer buffer size.

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the `spark.props` section and add the following setting. Modify 2000 (2GB) depending on whether your import is successful:

```
"spark.kryoserializer.buffer.max.mb": "2000"
```

3. Save your changes and restart the platform.
4. Attempt to import the dataset again. If it fails, you can try incrementally raising the above value.

For more information on passing property values into Spark, see *Configure for Spark*.

Using Databricks Table Connections

Uses of Databricks tables

The Designer Cloud powered by Trifacta platform can use Databricks Tables for the following tasks:

1. Create datasets by reading from Databricks Tables tables.
2. Write data to Databricks Tables.

Table Type	Support	Notes
Databricks managed tables	Read /Write	
Delta tables	Read /Write	NOTE: Versioning and rollback of Delta tables is not supported within the Designer Cloud powered by Trifacta platform . The latest version is always used. You must use external tools to manage versioning and rollback.

External tables	Read /Write	NOTE: When writing to an external table the TRUNCATE and DROP publishing actions are not supported.
Databricks unmanaged tables	Read /Write	
Delta Tables (managed and unmanaged tables)	Read /Write	
Partitioned tables	Read	

The underlying format for Databricks Tables is Parquet.

Before you begin using Databricks tables

- **Databricks Tables deployment:** Your Trifacta administrator must enable use of Databricks Tables.
- **Databricks Personal Access Token:** You must acquire and save a Databricks Personal Access Token into your Trifacta account. For more information, see *Databricks Settings Page*.

Storing data in Databricks tables

NOTE: The Designer Cloud powered by Trifacta platform does not modify source data in Databricks Tables. Datasets sourced from Databricks Tables are read without modification from their source locations.

Reading from Databricks Tables

You can create a Trifacta dataset from a table or view stored in Databricks Tables.

- Read support is also available for Databricks Delta Lake.

NOTE: Custom SQL queries are supported. Multi-statement custom SQL is not supported for Databricks Tables. Custom SQL queries must be a single `SELECT` statement. For more information, see *Create Dataset with SQL*.

For more information on how data types are imported from Databricks Tables, see *Databricks Tables Data Type Conversions*.

Writing to Databricks Tables

You can write data back to Databricks Tables using one of the following methods:

- Job results can be written directly to Databricks Tables as part of the normal job execution.
 - Data is written as a managed table to DBFS in Parquet format.
 - Create a new publishing action to write to Databricks Tables. See *Run Job Page*.
- For more information on how data is converted to Databricks Tables, see *Databricks Tables Data Type Conversions*.

Ad-hoc Publishing to Databricks Tables

Not supported.

Reference

Supported Versions: n/a

Supported Environments:


Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Not supported	Supported	Supported
Write	Not supported	Supported	Supported

Tip: It's easier to create a connection of this type through the UI. Typically, only one connection is needed.

SFTP Connections

Contents:

- Limitations
- Prerequisites
 - SSH Keys
 - Whitelist SFTP server
- Enable
 - Configure file storage protocols and locations
 - Enforce authentication methods
 - Java VFS service
- Create Connection
 - Create through application
 - Create through APIs

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to SFTP servers to upload your datasets to the Designer Cloud application .

Linux- and Windows-based SFTP servers are supported.

Jobs can be executed from SFTP sources on the following running environments:

- HDFS-based Spark
- Azure Databricks
- Trifacta Photon
- Spark on EMR

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Limitations

- Files and folders with spaces or special characters in them cannot be used. For example, a file or folder on the SFTP server with a hashtag (#) in it cannot be used for data.
 - Files and folders whose names begin with underscore () are not visible.
- Ingest of over 500 files through SFTP at one time is not supported.
- You cannot run Spark jobs using Avro or Parquet sources uploaded via SFTP .
- You cannot publish compressed Snappy files to SFTP destinations.
- You cannot publish Hyper format to SFTP destinations.

Prerequisites

- Acquire user credentials to access the SFTP server. You can use username/password credentials or SSH keys. See below.
- Verify that the credentials can access the proper locations on the server where your data is stored. Initial directory of the user account must be accessible.

SSH Keys

If preferred, you can use SSH keys to for authentication to the SFTP server.

NOTE: SSH keys must be private RSA keys. If you have OpenSSH keys, you can use the ssh-keygen utility to convert them to private RSA keys.

Whitelist SFTP server

If you are running jobs on EMR or Azure Databricks , you must add the SFTP server to the whitelist of IPs that are permitted to communicate with the cluster. For more information, please see the documentation that is provided with your software distribution.

You must also add the SFTP server to the whitelisted of file storage systems. Details are below.

Enable

By default, this connection type is automatically enabled for use.

NOTE: You must provide the protocol identifier and storage locations for the SFTP server. See below.

Configure file storage protocols and locations

The Designer Cloud powered by Trifacta platform must be provided the list of protocols and locations for accessing SFTP .

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameters and set their values according to the table below:

```
"fileStorage.whitelist": ["sftp"],  
"fileStorage.defaultBaseUri": ["sftp:///"],
```

Parameter	Description
filestorage.whitelist	<div>A comma-separated list of protocols that are permitted to access SFTP . <div>NOTE: The protocol identifier "sftp" must be included in this list.</div></div>
filestorage.	

defaultBaseUri	<p>For each supported protocol, this parameter must contain a top-level path to the location where platform files can be stored. These files include uploads, samples, and temporary storage used during job execution.</p> <div> <p>NOTE: A separate base URI is required for each supported protocol. You may only have one base URI for each protocol.</p> </div> <div> <p>NOTE: For SFTP , three slashes at the end are required, as the third one is the end of the path value. This value is used as the base URI for all SFTP connections created in Designer Cloud Powered by Trifacta Enterprise Edition.</p> </div> <p>Example:</p> <pre>sftp:///</pre> <p>The above example is the most common example, as it is used as the base URI for all SFTP connections that you create. If you add a server value to the above URI, you limit all SFTP connections that you create to that specified server.</p>
----------------	---

3. Save your changes and restart the platform.

Enforce authentication methods

By default, the Designer Cloud application enables use of two different authentication mechanisms:

- Basic - use a password to access the SFTP server
- SSHKey - use a public SSHKey and password to access the SFTP server

Along with basic and SSH key, the SFTP servers in your environment may be configured with other authentication methods, and those methods sometimes take precedence. As a result, when using default authentication methods, SFTP connections from the Designer Cloud powered by Trifacta platform can fail to connect to the SFTP server.

To eliminate these issues, you can configure the Designer Cloud application to enforce usage of one of the following authentication schemes. These schemes are passed to the SFTP server during connection time, which forces the server to use the appropriate method of authentication. When the following parameter is specified, SFTP connections can be configured using the listed methods and should work for connecting to the server.

NOTE: Enforcement applies to connections created via the APIs as well. After configuration, please be sure to use one of the enforced authentication methods when configuring your SFTP connections through the application or the APIs.

Steps:

1. To apply this configuration change, login as an administrator to the Trifacta node. Then, edit `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameter in the configuration file:

```
"batchserver.workers.filewriter.hadoopConfig.sftp.PreferredAuthentications"
```

3. Set the parameter value according to the following:

Preferred authentication method	Parameter value	Description
Basic	"password"	<p>Basic password authentication method is used to connect to the SFTP server.</p> <div> <p>NOTE: You must configure your SFTP server connection in the platform to use the Basic method.</p> </div>

SSHKey	"publickey"	SSH Key authentication method is used.
		NOTE: You must configure your SFTP server connection in the platform to use the SSHKey method.
both	"publickey, password"	Both methods of authentication are supported.

4. Save your changes and restart the platform.

Java VFS service

Use of SFTP connections requires the Java VFS service.

NOTE: This service is enabled by default.

For more information on configuring this service, see *Configure Java VFS Service*.

Create Connection

Create through application

You can create a SFTP connection through the Designer Cloud application .

Steps:

1. In the left nav bar, select the Connections icon. See *Connections Page*.
2. In the Connections page, click **Create Connection**. See *Create Connection Window*.
3. In the Create Connection window, click the SFTP connection card.
4. Specify the properties for your SFTP server.

Property	Description
Host	The hostname of the FTP server to which you are connecting. Do not include any protocol identifier (<code>sftp://</code>).
Port	The port number to use to connect to the server. Default port number is 22.
Credential Type	Select one of the following: basic - authenticate via username and password SSH Key - authenticate via username and SSH key
User Name	The username to use to connect.
Password	(Basic credential type) The password associated with the username.
SSH Key	(SSH Key credential type) The SSH key that applies to the username.
Test Connection	Click this button to test the connection that you have specified.
Default Directory	Absolute path on the SFTP server where users of the connection can begin browsing.
Block Size (Bytes)	Fetch size in bytes for each read from the SFTP server.
	NOTE: Raising this value may increase speed of read operations. However, if it is raised too high, resources can become overwhelmed, and the read can fail.

Connection Name	The name of the connection as you want it to appear in the application.
Description	This description is displayed in the application.

For more information, see *Create Connection Window*.

5. Click **Save**.

Create through APIs

- Type: `jdbc`
- Vendor: `sftp`

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

REST API Connections

Contents:

- *Limitations*
 - *Prerequisites*
 - *Enable connections on the same cluster*
 - *Configure*
 - *Authentication types*
 - *Configure endpoints*
 - *Connect string options*
 - *Create via API*
 - *Example - single GET method*
 - *Example - rate limiting*
 - *Use*
 - *Using REST API Connections*
 - *Uses*
 - *Before you begin*
 - *Secure access*
 - *Reading data*
 - *Writing data*
 - *Reference*
-

The REST API connection type provides a generic interface to relational data available through REST APIs. Using this connection type, you can create connections to individual endpoints across hundreds of REST-based applications.

Limitations

- Import-only connection type
- A limited set of request methods is supported. See Method entry below.
- Using a passphrase when generating an SSH key is not supported.
- JSON response from API endpoint is required. API endpoints that return XML responses are not supported.
- OAuth 2.0 authentication is not supported.
- After the initial connection to an endpoint is made, the schema is cached. The schema is not updated again until the connection is edited.
- By default, the number of endpoints that you can specify to use an individual connection is 10. This limit can be modified.
 - For more information, see *Workspace Settings Page*.

Prerequisites

- You should identify the tables and (optional) data models for them that you wish to access.
- You should acquire the credentials to access your target endpoints for one of the supported authentication methods.
 - If you are using a key or token to access the endpoints, you should generate this token before you begin.
 - See below.

Enable connections on the same cluster

For security reasons, the Designer Cloud application prevents connections from services external to the product from within the same cluster by default. If you are creating connections to REST applications that are hosted on the same cluster as the Trifacta node, additional configuration is required. Please complete the following configuration steps.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameters and populate them as described below:

Parameter	Description
<code>data-service.allowedUrlRegexes</code>	<p>This parameter defines an array list of regular expressions for URLs that are permitted to connect to the data service. Please add the URL of the REST application to which you are connecting to the array list.</p> <p>The following example permits URLs that include <code>localhost</code> and <code>example</code>:</p> <pre>["localhost", "example"]</pre>
<code>data-service.forbiddenIPs</code>	<p>This parameter defines the list of IP addresses that are prevented from connecting to the data service. Please remove the following from this list:</p> <pre>10.0.0.0/8</pre>

3. Save your changes and restart the platform.

Configure

To create this connection, in the Connections page , select the Applications tab. Click the REST API card. See *Connections Page*.

Modify the following properties as needed:

Property	Description
Base URI	<p>The base URI for the endpoints to which you wish to connect through this connection. Example:</p> <pre>https://exampleserver.sharepoint.com/sites/SharePointTest</pre> <p>Tip: SSL access is supported over the HTTPS protocol.</p>
Connect String Options	<p>Apply any connection string options that are part of your authentication to REST API .</p> <p>A default string has been provided for you. For more information, see below.</p>
Authentication Type	The method by which you wish to authenticate to the endpoint. See "Authentication types" below.
API endpoints	Specify the endpoints to which to connect. For more information, see "Configure endpoints" below.
Test Connection	After you have defined the REST API credentials and connection string, you can validate those credentials.
Connection Name	Display name of the connection

Connection Description	Description of the connection, which appears in the application.
------------------------	--

Authentication types

The following types of authentication are supported for REST API connections. For each type, additional properties may require configuration.

Basic auth

A username/password combination is submitted as part of any request for authentication.

Property	Description
Username	Username to access the endpoints.
Password	Password associated with the username.

HTTP Header Based Auth

Authentication is submitted using a key/value pair submitted in the HTTP request header.

Property	Description
Header Key	Key for the header parameter used in authentication
Header Value	Credential associated with header authentication key

HTTP Query Based Auth

Authentication is submitted using a query parameter key/value pair submitted as part of the URL.

Property	Description
Query Key	Key for the query parameter used in authentication
Query Value	Credential associated with the query parameter authentication key

Configure endpoints

Each endpoint and method combination must be configured. To add an endpoint, click **Add endpoint**.

The properties are described below.

Property	Description
Method	<p>API request method to use. Supported methods:</p> <ul style="list-style-type: none"> • GET - read from the endpoint • POST - create a new instance of an object in the target application through the endpoint • PUT - modify an existing instance <div style="border: 1px solid red; padding: 10px; margin: 10px 0;"> <p>In some target systems, the PUT and POST methods are required for generating datasets for import. These methods should not be used for other uses cases, such as writing data back to the target system. REST API connections are supported for import only.</p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>NOTE: Other methods are not supported for use.</p> </div>

URL Endpoint	<p>The endpoint that you are accessing using the specified method.</p> <div> Tip: The Base URI value and this value should form a complete URL. </div>
Table Name	(required) The name of the table with which you are interacting through this endpoint.
Data Model	<p>Select the type of model used for the selected table:</p> <ul style="list-style-type: none"> • Document - Data is returned as a document containing top-level elements which are represented as columns in the Designer Cloud application . Nested data is returned in aggregated form. • Relational - Data is returned in tabular form, in which each returned XPath represents an individual table containing a primary key and a foreign key linking to the parent document. • Flattened Documents - Data is stored as FlattenedArrays in the source system. A separate table is returned for each object array and is joined to its parent table. The parent table and each child table are joined into a single table for use in the Designer Cloud application . <p>For more information on these data model types, see https://cdn.cdata.com/help/DWE/jdbc/pg_RESTParsing.htm.</p>
Pagination	Select the type of pagination to request to the API endpoint. See "Pagination" below.
Advanced options: Custom Header	<p>(optional) You can insert a custom header in the request as a key/value pair.</p> <p>To add more headers, click Add.</p>
Advanced options: Query Parameter	<p>(optional) You can append a query parameter and value to the URL. These values are appended in the following form:</p> <div> <code><endpoint_url>?<key1>=<value1>&<key2>=<value2></code> </div> <p>To add more query parameters, click Add.</p>
Advanced options: XPath	(optional) You can specify an XPath to be queried of the URL.

Pagination

For the selected endpoint, you can specify the type of pagination in use by the target application. Specifying the pagination allows the Designer Cloud application to retrieve larger sets of records when pagination is in use. For more information on these pagination methods, see http://cdn.cdata.com/help/DWE/ado/pg_customschemaselect.htm.

None:

(default) No pagination is applied by the endpoint.

Next page URL:

When this method is selected, the URL of the next page of results is returned as part of the response body.

- **Page URL path** defines the XPath in the response to the attribute containing the URL of the next page.

Paging token:

A paging token may be returned as part of the response. To acquire the next page of results, this token must be submitted in the subsequent request as the value associated with the paging parameter.

- **Page token path** is the XPath to the token that must be submitted with the next request.
- **Page token param** is the parameter in the request into which the page token must be submitted.
- **More pages param** (optional) is used when the page token path must be submitted as a query parameter. This value defines the query parameter for which it is submitted.

Record offset:

Under this pagination method, subsequent pages of results can be queried based on defining the number of results (records) to offset with the query.

- **Page offset param** defines the query parameter where you can specify the page offset to query.
- **Page size param** defines the parameter in the request where you define the size in records of each request (page) of records.
- **Page size** defines the number of records to request in a page.

Page number:

Similar to record offset, this method queries results based on specified page numbers.

- **Page number param** defines the query parameter where you can specify the page number to query.
- **Page size param** defines the parameter in the request where you define the size in records of each page of records.
- **Page size** defines the number of records to request in a page.

Connect string options

Too many requests error

During execution, you may receive an error similar from the driver to the following:

```
PlatformErrors: Retries errors based on the exception message. E.g. Other=PlatformErrors="Too Many Requests"
MaximumRequestRetries: The number of times the driver will attempt to retry the request (Default 4)
```

In this case, the default wait time for retrying a request (2 seconds) is not enough time, and the requests are piling up. You can address this issue by inserting the following connect string options:

```
Other='RetryWaitTime=15000'
```

The above option sets the wait time before retrying to 15000ms (15 seconds). You can experiment with this value as needed.

For more information on available connect string options, see <https://cdn.cdata.com/help/DWE/ado/Connection.htm>.

Create via API

This connection can also be created using the API.

- Type: jdbc_rest
- Vendor: jdbc_rest

Example - single GET method

The following example request creates a REST API connection with the following characteristics:

- Query parameters are used for authentication
- A single endpoint is enabled:
 - Method: GET
 - Target: table1
 - dataModel: Document

```
{
  "vendor": "jdbc_rest",
  "vendorName": "jdbc_rest",
```



```

"name": "REST API test",
"description": "",
"type": "jdbc",
"params": {
  "base_URI": "some base URI",
  "connectStrOpts": ""
},
"credentialType": "httpQueryBasedAuth",
"credentials": [
  {
    "queryKey": "user",
    "queryValue": "token"
  }
],
"endpoints": [
  {
    "tableName": "table1",
    "httpMethod": "get",
    "endpoint": "endpoint1",
    "requestBody": "",
    "xPath": "",
    "dataModel": "document",
    "headers": {},
    "queryParams": {}
  }
]
}

```

Example - rate limiting

The following example creates a REST API connection to polygon.io with the following characteristics:

- Query-based authentication using key/value pair
- Connect String Options:

```
Other='RetryWaitTime=15000';
```

- Wait for retry: 15000 milliseconds
- Single endpoint to GET stock ticker information:
 - DataModel: Document
 - XPath: \$. /results
 - Rate limiting on the endpoint (maximum queries per minute): 1000
 - queryParams.date is a parameter that is passed in for this specific connection type.
 - Pagination: nextPageURL method

```

{
  "vendor": "jdbc_rest",
  "vendorName": "jdbc_rest",
  "name": "REST API",
  "description": "",
  "type": "jdbc",
  "params": {
    "base_URI": "https://api.polygon.io/",
    "connectStrOpts": "Other='RetryWaitTime=15000';"
  },
  "credentialType": "httpQueryBasedAuth",
  "credentials": [
    {
      "queryKey": "apiKey",
      "queryValue": "someKey"
    }
  ],
  "endpoints": [

```

```

{
  "tableName": "tickers",
  "httpMethod": "get",
  "endpoint": "/v3/reference/tickers",
  "requestBody": "",
  "xPath": "$./results",
  "dataModel": "document",
  "headers": {},
  "queryParams": {
    "limit": "1000",
    "date": "2021-11-04T00:00:00Z"
  },
  "pagination": {
    "pageurlpath": "$./next_url",
    "paginationType": "nextPageURL"
  }
}
]
}

```

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Use

You can import datasets from REST API through the Import Data page. See *Import Data Page*.

Tip: You can perform joins and unions using custom SQL as part of your initial request for data. It may be easier to import the tables as separate datasets and then to perform the join or union within the Designer Cloud application .

Using REST API Connections

This section describes how you interact through the Designer Cloud powered by Trifacta® platform with your REST data.

Uses

The Designer Cloud powered by Trifacta platform can use REST API connections for the following tasks:

1. Import datasets

Before you begin

- **Read Access:** You must have credentials to create access the specific endpoints required to retrieve your data.
- **Write Access:** Not supported

Secure access

SSL is available over HTTPS for REST API connections.

Reading data

You can create a Trifacta dataset from the following data models:

1. documents
2. flattened documents

3. relational tables

For more information, see *Database Browser*.

Writing data

Not supported.

NOTE: Do not use the `PUT` and `POST` methods to write data back into the target system.

Reference

Supported Environments:

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Not supported	Not supported	Not supported

Enable Teradata Access

Contents:

- *Limitations*
- *Create Teradata Connection*
 - *Troubleshooting*
- *Testing*

This section provides information on how to enable connection to Teradata .

- Teradata provides Datawarehousing & Analytics solutions and Marketing applications. The Teradata supports all of their Data warehousing solutions. For more information, see <http://www.teradata.com>.
- For more information on supported versions, see *Connection Types*.

This connection supports reading and writing. You can create multiple Teradata connections in the Designer Cloud application .

Supported Versions: 14.10+

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Not supported
Write	Supported	Supported	Supported

Limitations

- By default, Teradata does not permit the publication of datasets containing duplicate rows. Workarounds:
 - Your final statement for any recipe that generates results for Teradata should include a `Remove duplicate rows` transformation.

NOTE: The above transformation removes exact, case-sensitive duplicate rows. Teradata may still prevent publication for case-insensitive duplicates.

- It's possible to change the default writing method to Teradata to enable duplicate rows. For more information, contact *Alteryx Support*.
- When creating custom datasets using SQL from Teradata sources, the `ORDER BY` clause in standard SQL does not work. This is a known issue.

Create Teradata Connection

For more information on creating a Teradata connection, see *Teradata Connections*.

Troubleshooting

Error	Description
Duplicate row error	<p>This error occurs when duplicate rows are being inserted during publishing to Teradata .</p> <p>Workaround: All inserted rows must be unique ,or the Teradata tables must be <code>MULTISET</code>. To configure Designer Cloud Powered by Trifacta Enterprise Edition to use <code>MULTISET</code> tables, please contact <i>Alteryx Support</i>.</p>

Testing

Steps:


1. After you create your connection, load a small dataset based on a table in the connected Teradata . See *Import Data Page*.
2. Perform a few simple transformations to the data. Run the job. See *Transformer Page*.
3. Verify the results.

For more information, see *Verify Operations*.

Teradata Connections

Contents:

- *Limitations*
- *Prerequisites*
- *Create Teradata Connection*
 - *Connection URL*
 - *Driver Information*
 - *Create via API*
- *Using Teradata Connections*
 - *Uses of Teradata*
 - *Before you begin using Teradata*
 - *Writing to Teradata*
 - *SQL Syntax*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to your Teradata instance from the Designer Cloud powered by Trifacta® platform .

Supported Environments:

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Limitations

- Tables inside the DBC database are not listed due to technical constraints; however you can access the tables through custom SQL.
- Custom SQL is supported. The Custom SQL should be in the following format:

```
SELECT * FROM <DATABASE_NAME>.<Table_NAME>;
```

- When using custom SQL, the keyword `LIMIT` is not supported by Teradata . Use the keyword `TOP` to get the required number of rows similar to the following:

```
SELECT TOP <NUMBER_OF_ROWS> * FORM <DATABASE_NAME>.<TABLE_NAME>;
```

- In the `Connect String Options`, the parameters must be separated by commas:

```
Database=DBC,DBC_PORT=1025
```

Prerequisites

- Additional setup is required. For more information, see *Enable Teradata Access*.

Create Teradata Connection

To create this connection, in the Connections page, select the Databases tab. Click the Teradata card. See *Connections Page*.

Modify the following properties as needed:

Property	Description
Host	Enter the host name of Teradata . Example value: <div>buick1.teradata.ws</div>
Port	Set this value to the port number through which to access Teradata . By default, this values is set to 1025.
Connect String Options	(Optional) You can specify additional options used to connect as a string value. Connect string options are submitted in the following format: <div>option1=value1,option2=value2</div>
Enable Data Encryption	When enabled, the data exchanged between the Teradata JDBC driver and the database is encrypted.
User Name	The username used to connect.
Password	The password associated with the username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can verify that the Designer Cloud application can use them to connect to the database.
Advanced options: Default Column Data Type Inference	Set to <code>disabled</code> to prevent the product from applying its own type inference to each column on import. The default value is <code>enabled</code> .
Advanced options: Enable SSH Tunneling	This feature is not available.
Connection Name	Display name of the connection.
Connection Description	(Optional) Description of the connection, which appears in the application.

Connection URL

The properties that you provide are inserted into the following URL, which connects the Designer Cloud powered by Trifacta platform to the connection:

```
jdbc:teradata://<host>/DBS_PORT=<port>,<connect-string-options>
```

where:

Parameter	Description
<host>	Host URL of the database server.
<port>	Port number for the database server. Typically, this value is 1025 for Teradata .
<connect-string-options>	Connect string options, which are submitted in the following format: <div></div>

```
option1=value1,option2=value2
```

The Connection URL is mostly built up automatically using cluster configuration for the platform.

Use Data Encryption

When Data Encryption is enabled for the connection, the following is automatically appended to the connect string options:

```
,ENCRYPTDATA=ON
```

Driver Information

This connection uses the following driver:

- **Driver name:** `com.teradata.jdbc.TeraDriver`
- **Driver version:** `com.teradata:terajdbc4:17.00.00.03`
- **Driver documentation:** <https://developer.teradata.com/connectivity/reference/jdbc-driver>

Create via API

This connection can also be created using the API.

```
"vendor": "teradata",  
"vendorName": "Teradata",  
"type": "jdbc"
```

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Using Teradata Connections

Uses of Teradata

The Designer Cloud powered by Trifacta platform can use for the following tasks:

- Create datasets by reading from tables.
- Writing back to Teradata .

Before you begin using Teradata

Read Access:

Your administrator must configure read permissions.

Writing to Teradata

Supported.

SQL Syntax

The following syntax requirements apply to this connection.

Object delimiter: double-quote or no delimiter

Example syntax:

Double quotes can be used around database names, table names, or column names.

Note that references to specific values must be single-quoted for columns of String data type.


```
SELECT "column1","column2" FROM "databaseName"."tableName" WHERE "column3" = 'my_value';
```

For more information on SQL in general, see *Supported SQL Syntax*.

MongoDB Connections

Contents:

- *Prerequisites*
- *Limitations*
- *Create Connection*
 - *MongoDB*
 - *MongoDB Atlas*
 - *Create connection via API*
 - *Connect string options*
- *Using MongoDB*
 - *MongoDB Data Organization Hierarchy*
 - *Database Uses*
 - *Read Data*
- *Data Type Mappings*
 - *Access/Read*
 - *Write/Publish*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to MongoDB and MongoDB Atlas connections through Designer Cloud application . These connections enable to read data from the MongoDB workspace.**Supported Versions:**

- Database versions: 2.6 - 6.0.1

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Not supported	Not supported	Not supported

Prerequisites

- MongoDB supports basic (username/password) authentication.

Limitations

NOTE: During normal selection or import of an entire table, you may encounter an error indicating a problem with a specific column. Since some tables require filtering based on a particular column, data from them can only be ingested using custom SQL statements. In this case, the problematic column can be used as a filter in the WHERE clause of a custom SQL statement to ingest the table.

- For more information, please consult the CData driver documentation for the specific table.
- For more information on using custom SQL, see *Create Dataset with SQL*.

NOTE: For filtering date columns, this connection type supports a set of literal functions on dates. You can use these to reduce the volume of data extracted from the database using a custom SQL query. For

more information, see the `pg_dateliteralfunctions.htm` page in the driver documentation for this connection type.

- This connection is read-only.

Create Connection

MongoDB

To create a MongoDB connection, please specify the following properties:

Property	Description
Host	Name of the host.
Port	Set this value to the port number through which to access MongoDB. By default, this value is 27017.
Database	The database that you want to read
Auth Database	Name of the MongoDB database used for authentication
Replica Set	(Optional) Comma-separated list of secondary servers in the replica set, specified by address and port. A replica set is a group of mongoDB processes that maintain the same data set. Replica sets provide redundancy and high availability and are the basis for all production deployments. For more information, see https://docs.mongodb.com/manual/replication/ .
Secondary Reads	Enable this checkbox if you want to read from secondary (slave) servers.
Use SSL	Enable this checkbox if you want to connect using SSL.
Connect String Options	(Optional) You can specify additional options used to connect as a string value. The following option sets the connection timeout in milliseconds: <div>Timeout=0;</div> The default value is 0, which disables connection timeouts. See below for more information.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can verify that the Designer Cloud application can use them to connect to the database.
Advanced options: Default Column Data Type Inference	Set to disabled to prevent the product from applying its own type inference to each column on import. The default value is enabled.
Advanced options: Enable SSH Tunneling	This feature is not available.
Connection Name	Display name of the connection
Connection Description	(Optional) Description of the connection, which appears in the application.

MongoDB Atlas

To create a MongoDB Atlas connection, please specify the following properties:

Property	Description
Host	Name of the host.
Port	Set this value to the port number through which to access MongoDB. By default, this value is 27017.
Database	The database that you want to read
Replica Set	(Optional) Comma-separated list of secondary servers in the replica set, specified by address and port.

	A replica set is a group of mongoDB processes that maintain the same data set. Replica sets provide redundancy and high availability and are the basis for all production deployments. For more information, see https://docs.mongodb.com/manual/replication/ .
Secondary Reads	Enable this checkbox if you want to read from secondary (slave) servers.
Connect String Options	<p>(Optional) The option sets the connection timeout in milliseconds:</p> <pre>Timeout=0;</pre> <p>The default value is 0, which disables connection timeouts. See below for more information.</p>
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can verify that the Designer Cloud application can use them to connect to the database.
Advanced options: Default Column Data Type Inference	<p>Set to disabled to prevent the product from applying its own type inference to each column on import. The default value is enabled.</p> <p>NOTE: SSH tunneling is not supported for MongoDB Atlas.</p>
Connection Name	Display name of the connection
Connection Description	(Optional) Description of the connection, which appears in the application.

For more information on these settings, see <https://cdn.cdata.com/help/DGH/jdbc/>.

Create connection via API

Depending on your product edition, you can create connections of this type.

MongoDB:

```
"vendor": "mongodb",
"vendorName": "MongoDB",
"type": "jdbc"
```

MongoDB Atlas:

```
"vendor": "mongodb_atlas",
"vendorName": "MongoDB Atlas",
"type": "jdbc"
```

<https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Connect string options

Connection timeout

By default, the supported driver applies a connection timeout to MongoDB of 0 seconds. As needed, you can modify the connection timeout through connect string options:

```
Timeout=<value_in_seconds>;
```

where:

<value_in_seconds> corresponds to the number of seconds for the time.

Flattening Documents

Documents can contain other documents, which enables the storage of nested data. You can control the flattening of nested objects and arrays through the CData driver through Connect String Options.

NOTE: Columns that have been flattened can be accessed or referenced using custom SQL queries. Additional information is below.

Flatten Objects:

By default, the CData driver flattens nested Objects. As needed, you can set FlattenObjects to `false` to disable this behavior.

For more information, see http://cdn.cdata.com/help/DGG/jdbc/RSBMongodb_p_FlattenObjects.htm.

Flatten Arrays:

By default, CData driver does not flatten Arrays.

- As needed, you can configure the number of elements that you want to have returned in your flattened arrays.
- To flatten all elements of all arrays, set FlattenArrays to `-1`.

For more information, see http://cdn.cdata.com/help/DGG/jdbc/RSBMongodb_p_FlattenArrays.htm.

Referencing flattened columns:

If you have flattened Objects or Arrays, you can reference these columns using square brackets in your custom SQL queries.

Example of flattened Object:

```
SELECT [address.city] FROM my_table;
```

Example of flattened Array:

```
SELECT * FROM my_table WHERE [hobbies.0]='cricket';
```

Driver Information

For more information on CData JDBC drivers, see <https://cdn.cdata.com/help/DGH/jdbc/>.

Using MongoDB

MongoDB is a NoSQL document database that provides high performance, availability, and scalability.

MongoDB Data Organization Hierarchy

MongoDb has a two-level data hierarchy:

```
+ Schema1
+ Collection1
```

```
+ Collection2
+ Schema2
+ Collection3
+ Collection4
```

- **Schema** roughly corresponds to a database.
- **Collection** roughly corresponds to a table.
 - A collection is composed of documents. A **Document** is a binary JSON representation of the fields and values of a row.

Database Uses

For more information on interacting with databases, see *Using Databases*.

Read Data

You can import datasets from MongoDB through the Import Data page. See *Import Data Page*.

Data Type Mappings

NOTE: The Trifacta® data types listed in this section reflect the raw data type of the converted column. Depending on the contents of the column, the application may re-infer a different data type, when a dataset using this type of source is loaded.

Access/Read

When data is imported from MongoDB, the supported data types from the source are converted to corresponding data types supported by the Designer Cloud application. For more information, see *Type Conversions*.

Source Data Type	Supported	Trifacta data type
ObjectId	Y	String
RegEx	Y	String
String	Y	String
Binary	Y	String
Integer	Y	Integer
Timestamp	Y	Datetime
Double	Y	Float
Array	Y	String
Bool	Y	bool
Null	Y	String
Date	Y	Datetime

Write/Publish

Not supported.

Tableau Server Connections

Contents:

- *Limitations*
- *Enable Hyper format*
- *Configure Permissions*
- *Create Tableau Server Connection*
 - *Create through application*
 - *Create through APIs*

This section describes the basics of creating Tableau Server connections from within the application.

NOTE: You can export Tableau Server files as part of exporting results from the platform. For more information, see *Publishing Dialog*.

Supported Versions: 10.5.x and later

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Not supported	Not supported	Not supported
Write	Supported	Supported	Supported

Limitations

- This connection type only enables publication.
 - You cannot read data from Tableau Server .
 - When created in the application, publish-only connections must be created through the Connections page.

Enable Hyper format

Hyper format generation is enabled by default. To enable the generation of results into Hyper format, please verify the following:

Steps:

1. Login as an administrator.
2. You apply this change through the *Workspace Settings Page*. For more information, see *Platform Configuration Methods*.
3. Locate the following setting:

Hyper output format

4. Set it to *Enabled*.
5. No other configuration is required.

Configure Permissions

The user who is publishing to Tableau Server must have exec permissions on the temporary directory on the backend datastore. This directory is used to write the intermediate file format locally, before it is published to Tableau Server. For more information, see *Supported File Formats*.

Create Tableau Server Connection

Create through application

Any user can create a Tableau Server connection through the application.

NOTE: Only an administrator can make a Tableau Server connection available for all users.

Steps:

1. In the left nav bar, select the Connections icon. See *Connections Page*.
2. In the Connections page, click **Create Connection**. See *Create Connection Window*.
3. In the Create Connection window, click the **Tableau Server** connection card.
4. Specify the properties for your Tableau Server.

Property	Description
Server URL	<div>The URL to the Tableau Server to which you are connecting. To specify an SSL connection, use <code>https://</code> for the protocol identifier.</div> <div>NOTE: By default, this connection assumes that the port number is 80. To use a different port, you must specify it as part of the Server name value: <code>http://<Tableau_Server_URL>:<port_number></code></div>
Site	<div>Enter the value that appears after <code>/site/</code> in your target location.</div> <div>Example target URL:</div> <div><code>https://tableau.example.com/#/site/MyNewTargetSite</code></div> <div>Enter the following for the Site setting:</div> <div><code>MyNewTargetSite</code></div>
User Name	The username to use to connect.
Password	The password associated with the username.
Test Connection	Click this button to test the connection that you have specified.
Connection Name	The name of the connection as you want it to appear in the user interface.
Description	This description is displayed in the user interface.

For more information, see *Create Connection Window*.

5. Click **Save**.

Create through APIs

You can create this connection type through the APIs:

API: *API Reference*

- Type: jdbc
- Vendor: tableau

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Salesforce Connections

Contents:

- *Limitations*
 - *Prerequisites*
 - *Enable*
 - *Configure*
 - *Connect to Sandbox account*
 - *Connect string options*
 - *Create via API*
 - *Use*
 - *Using Salesforce Connections*
 - *Uses of Salesforce*
 - *Before you begin using Salesforce*
 - *Secure access*
 - *Storing data in Salesforce*
 - *Reading from Salesforce*
 - *Writing to Salesforce*
 - *Reference*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to your Salesforce instance from the Designer Cloud powered by Trifacta® platform . This connector is designed as a wrapper around the Salesforce REST API.

Limitations

NOTE: During normal selection or import of an entire table, you may encounter an error indicating a problem with a specific column. Since some tables require filtering based on a particular column, data from them can only be ingested using custom SQL statements. In this case, the problematic column can be used as a filter in the WHERE clause of a custom SQL statement to ingest the table.

- For more information, please consult the CData driver documentation for the specific table.
- For more information on using custom SQL, see *Create Dataset with SQL*.

NOTE: For filtering date columns, this connection type supports a set of literal functions on dates. You can use these to reduce the volume of data extracted from the database using a custom SQL query. For more information, see the `pg_dateliteralfunctions.htm` page in the driver documentation for this connection type.

- This is a read-only connection.
- Single Sign-On (SSO) is not supported.
- Custom domains are not supported.
- You cannot ingest Salesforce tables that require mandatory filters.

Prerequisites

- The account used to login from the Designer Cloud powered by Trifacta platform must access Salesforce through a security token.

NOTE: Please contact your Salesforce administrator for the Server Name and the Security Token values.

- The logged-in user must have required access to the tables and schema.
- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Enable

- General relational connectivity must be enabled. For more information, see *Relational Access*.
- This connection type utilizes OAuth 2.0 for authentication.

NOTE: OAuth 2.0 authentication requires additional configuration specific to the connection type.

For more information, see *Enable OAuth 2.0 Authentication*.

Configure

To create this connection, in the Connections page, select the Applications tab. Click the Salesforce card. See *Connections Page*.

Modify the following properties as needed:

Property	Description
Server Name	Enter the host name of your Salesforce implementation. Example value: <div>exampleserver.salesforce.com</div>
Connect String Options	Apply any connection string options that are part of your authentication to Salesforce. For more information, see below.
Credential Type	Select the type of credentials to provide with the connection: <ul style="list-style-type: none">• <code>SecurityToken</code> - apply the security token that has been generated within the account to authenticate to Salesforce.• <code>OAuth 2.0</code> - use OAuth 2.0 client connect to Salesforce. An OAuth 2.0 client may already be defined in the Designer Cloud application . <div>NOTE: After you have specified the connection to use OAuth 2.0, click Authenticate to validate the connection with the target datastore. If you have modified the connection, click Re-authenticate to validate the new connection definition. You must re-authenticate if you receive an expired tokens message.</div>
OAuth 2.0 Client	(OAuth 2.0 credential type) Select the OAuth 2.0 client to use.
User Name	(SecurityToken credential type) Username to use to connect to the database.

Password	(SecurityToken credential type) Password associated with the above username.
Security Token generated in account	(SecurityToken credential type) Paste the security token associated with the account to use for this connection.
Test Connection	(SecurityToken credential type) After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Default Column Data Type Inference	Set to disabled to prevent the platform from applying its own type inference to each column on import. The default value is enabled .
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connect to Sandbox account

If you are connecting to a Salesforce sandbox account, the following property values must be modified:

Property	Sandbox Value
Server Name	<div>test.salesforce.com</div>
Connect String Options	<p>Please append the following to your connect string options:</p> <div>UseSandbox=true;</div> <p>For more information, see below.</p>
OAuth 2.0 client	Select Salesforce Sandbox.
User Name	<p>(Security token credential type) Append the name of the sandbox environment to the end of the username. In the following, the sandbox environment is demo:</p> <div>exampleUser@example.com.demo</div>

For OAuth 2.0 authentication:

The OAuth 2.0 client must be modified to use the following values from test.salesforce.com:

Property	Setting
Authorization URL	<div>"https://test.salesforce.com/services/oauth2/authorize"</div>
Token Url	<div>"https://test.salesforce.com/services/oauth2/token"</div>

Connect string options

Connection timeout

By default, the supported driver applies a connection timeout to Salesforce of 60 seconds. As needed, you can modify the connection timeout through connect string options:

```
timeout=<value_in_seconds>
```

where:

<value_in_seconds> corresponds to the number of seconds for the time.

NOTE: Although it is not recommended, you can set this value to 0 to disable timeouts.

Schema caching

By default, the connection driver uses schema caching to speed up ingestion. To surface changes to Salesforce tables/schema immediately, you can use the following options to disable schema caching by the connection:

```
Other='cachemetadatatable=false;cachemetadatatablecolumns=false;'
```

Create via API

This connection can also be created using the API.

NOTE: If you are using OAuth 2.0 authentication for this type, you cannot create connections via API.

- Type: jdbc
- Vendor: salesforce

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Use

You can import datasets from Salesforce. See *Database Browser*.

Using Salesforce Connections

Uses of Salesforce

The Designer Cloud powered by Trifacta platform can use Salesforce for the following tasks:

1. Create datasets by reading from Salesforce tables.

Before you begin using Salesforce

Read Access:

- Your Salesforce administrator must configure read permissions.
- You must acquire a Salesforce security token for use with the Salesforce connection.

Secure access

SSL is the default connection method.

Storing data in Salesforce

Your Salesforce administrator should provide database access for storing datasets. Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: The Designer Cloud powered by Trifacta platform does not modify source data in Salesforce. Datasets sourced from Salesforce are read without modification from their source locations.

Reading from Salesforce

When the Designer Cloud powered by Trifacta platform connects to your Salesforce instance, the application can read from all Salesforce objects that are accessible through the Salesforce account in use, including:

- Salesforce objects and fields are mapped to tables and columns
- Standard and custom objects

NOTE: The names of custom objects are appended with the value `_c`.

- Audit columns
- System fields

NOTE: Unquoted identifiers are converted to uppercase during import.

You can create a Trifacta dataset from a table stored in Salesforce.

Writing to Salesforce

Not supported.

Reference

Supported Versions: n/a

Supported Environments:

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Not supported	Not supported	Not supported

SharePoint Connections

Contents:

- *Limitations*
 - *Prerequisites*
 - *Enable*
 - *Configure*
 - *Connect string options*
 - *Create via API*
 - *Use*
 - *Using SharePoint Connections*
 - *Uses of SharePoint*
 - *Before You Begin Using SharePoint*
 - *Secure access*
 - *Storing data in SharePoint*
 - *Reading from SharePoint*
 - *Writing to SharePoint*
 - *Reference*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can create connections to your Microsoft SharePoint instance from the Designer Cloud powered by Trifacta® platform .

You can create connections to:

- SharePoint On-Premises installations in your enterprise infrastructure
- SharePoint Online

NOTE: This connection supports reading from and writing to SharePoint lists.

For more information on Microsoft SharePoint , see <https://www.microsoft.com/en-us/microsoft-365/sharepoint/collaboration>.

Limitations

NOTE: During normal selection or import of an entire table, you may encounter an error indicating a problem with a specific column. Since some tables require filtering based on a particular column, data from them can only be ingested using custom SQL statements. In this case, the problematic column can be used as a filter in the WHERE clause of a custom SQL statement to ingest the table.

- For more information, please consult the CData driver documentation for the specific table.
- For more information on using custom SQL, see *Create Dataset with SQL*.

NOTE: For filtering date columns, this connection type supports a set of literal functions on dates. You can use these to reduce the volume of data extracted from the database using a custom SQL query. For more information, see the `pg_dateliteralfunctions.htm` page in the driver documentation for this connection type.

- Single Sign-On (SSO) is not supported.
- Column names are not validated on publishing.
- The SharePoint connection uses SharePoint APIs. As a result, transaction management and rollbacks are not supported.
- No schema validation is performed as part of writing results to SharePoint Lists.

Prerequisites

- The logged-in user must have required access to the tables and schema.
- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Enable

- General relational connectivity must be enabled. For more information, see *Relational Access*.

Configure

To create this connection, in the Connections page, select the Applications tab. Click the **SharePoint** card. See *Connections Page*.

Modify the following properties as needed:

Property	Description
SharePoint URL	<p>Enter the URL for your SharePoint Site or sub-site. Example value:</p> <div><code>https://exampleserver.sharepoint.com/sites/SharePointTest</code></div>
SharePoint Edition	<p>Product edition of SharePoint in use:</p> <ul style="list-style-type: none">• <code>SharePointOnPremise</code> - Use this option if you have an on-premises installation of SharePoint to which you can connect within your enterprise infrastructure.• <code>SharePointOnline</code> - Use this option if you are connecting to SharePoint Online. <div>NOTE: OAuth 2.0 for SharePoint is supported for SharePointOnline only.</div>
Auth Scheme	<p>Authentication scheme:</p> <ul style="list-style-type: none">• <code>Basic</code> - (for SharePointOnline) username and password• <code>NTLM</code> - (for SharePointOnPremise) Windows-based authentication scheme for on-premises deployments.• <code>OAuth</code> - (for SharePointOnline) Use OAuth 2.0 authentication to access SharePoint Online. <div>NOTE: For OAuth 2.0 authentication, additional configuration is required. See <i>OAuth 2.0 for SharePoint</i>.</div> <ul style="list-style-type: none">• <code>Azure AD</code> - You can leverage Azure AD credentials for authenticating to SharePoint. No additional configuration is required.

Schema	For Basic Auth integration, this value must be set to SOAP. For OAuth 2.0 integration, this value must be set to REST.
User Name	Username to use to connect to SharePoint .
Password	Password associated with the above username.
Test Connection	After you have defined the SharePoint Edition, credentials, and connection string, you can validate those credentials.
Additional Connect String Options	Apply any connection string options that are part of your authentication to SharePoint . A default string has been provided for you. For more information, see below.
Default Column Data Type Inference	Set to <code>disabled</code> to prevent the platform from applying its own type inference to each column on import. The default value is <code>enabled</code> .
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Connect string options

The following connection string is provided for you:

```
AutoCache=false;CacheMetadata=false;CacheTolerance=1;timeout=0;ShowPredefinedColumns=false
```

Parameter	Description
AutoCache	When enabled, the connection leverages any data that is automatically cached for each table. The default is <code>false</code> .
CacheMetadata	When enabled, table metadata can be retrieved from the SharePoint cache. The default is <code>false</code> .
CacheTolerance	This setting defines the duration in hours that objects are permitted to live in the cache. The default is <code>1</code> .
timeout	This setting defines the number of seconds that a query to the SharePoint database is allowed to run without a response. The SharePoint default timeout is <code>60</code> , which may cause complex queries of larger datasets to timeout. The default value in the Connect String Options is <code>0</code> , which means that there is no enforced timeout. Other timeouts may apply.
ShowPredefinedColumns	When enabled, users of the connection are permitted to view the columns that are created with the table, such as Created By and Modified By columns. The default is <code>false</code> .

For more information, see <http://cdn.cdata.com/help/RSG/jdbc/Connection.htm>.

Create via API

This connection can also be created using the API.

- Type: `jdbc`
- Vendor: `sharepoint`

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Use

You can import datasets from SharePoint . See *Database Browser*.

Using SharePoint Connections

This section describes how you interact through the Designer Cloud powered by Trifacta® platform with your SharePoint Lists.

- SharePoint is a content management system for sharing and collaborating across the enterprise. For more information, see <https://sharepoint.microsoft.com>.
- In SharePoint, data is stored in an object called a List. Lists can also include non-tabular data.

NOTE: Non-tabular data and some SharePoint List columns are converted to strings on import. For more information, see *SharePoint Data Type Conversions*.

Uses of SharePoint

The Designer Cloud powered by Trifacta platform can use SharePoint for the following tasks:

1. Import datasets by reading from SharePoint Lists.
2. Write to SharePoint Lists with your job results.

Before You Begin Using SharePoint

- **Read Access:** Your SharePoint administrator must configure read permissions.
- **Write Access:** You can write and publish jobs results to SharePoint.

Secure access

SSL is available over HTTPS for SharePoint connections.

Storing data in SharePoint

Your SharePoint administrator should provide database access for storing datasets. Users should know where shared data is located and where personal data can be saved without interfering with or confusing other users.

NOTE: The **Designer Cloud powered by Trifacta platform** does not modify source data in SharePoint. Datasets sourced from SharePoint are read without modification from their source locations.

Reading from SharePoint

You can create a Trifacta dataset from a List stored in SharePoint.

NOTE: Reading data is supported for SharePoint Lists only.

For more information, see *Database Browser*.

Writing to SharePoint

You can write back data to SharePoint using one of the following methods:

- Job results can be written directly to SharePoint as part of the normal job execution. Create a new publishing action to write to SharePoint.
- For more information on how data is converted to SharePoint, see *SharePoint Data Type Conversions*.

Data Validation issues:

NOTE: Some Trifacta data types do not map exactly to SharePoint List data types. These differences may appear when writing to a new SharePoint List. For more information, see *SharePoint Data Type Conversions*.

NOTE: Column name validation is not supported.

- No validation is performed for the connection and any required permissions during job execution. So, you can be permitted to launch your job even if you do not have sufficient connectivity or permissions to access the data. The corresponding publish job fails at runtime.
- Prior to publication, no validation is performed on whether a target is a table or a view, so the job that was launched fails at runtime.

Reference

Supported versions: n/a

Supported Environments:

Operation	Designer Cloud powered by Trifacta platform	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Supported	Supported	Supported

Google Analytics Connections

Contents:

- *Limitations and Requirements*
- *Create Connection*
 - *via Designer Cloud application*
 - *Connect String Options*
- *Data Type Conversions*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Google Analytics is a web analytics service offered by Google that tracks and reports website traffic. For more information, see <https://analytics.google.com/analytics/web/provision/#/provision>.

Limitations and Requirements

NOTE: During normal selection or import of an entire table, you may encounter an error indicating a problem with a specific column. Since some tables require filtering based on a particular column, data from them can only be ingested using custom SQL statements. In this case, the problematic column can be used as a filter in the WHERE clause of a custom SQL statement to ingest the table.

- For more information, please consult the CData driver documentation for the specific table.
- For more information on using custom SQL, see *Create Dataset with SQL*.

NOTE: For filtering date columns, this connection type supports a set of literal functions on dates. You can use these to reduce the volume of data extracted from the database using a custom SQL query. For more information, see the `pg_dateliteralfunctions.htm` page in the driver documentation for this connection type.

NOTE: Most interactions with the Google Analytics datastore are formed as custom SQL queries. More information on syntax and examples is provided below.

- OAuth 2.0 authentication is required.
 - An OAuth 2.0 web client must be available for use in the Designer Cloud application . For more information, see *OAuth 2.0 for Google Analytics*.
 - You cannot create OAuth 2.0 connections via API.
- The following schema is supported: `UniversalAnalytics`.

Tip: This schema was previously called `GoogleAnalytics` by the driver vendor.

- Google Analytics allows up to 10 metrics and seven dimensions in a single query.
 - When issuing a query that selects all columns, only the default Metric columns are selected for tables with more than 10 Metrics.
 - The default Dimensions are used unless you explicitly select other dimension columns.

- All reports in Google Analytics must cover a specific date range.
 - The default behavior is to pull the last month of data if the StartDate and EndDate inputs are left unset.
 - To override this behavior, the values can be set directly in the query.

Create Connection

via Designer Cloud application

When you create the connection, please review the following properties and specify them accordingly:

Connection Property	Description
View Id	Unique identifier of the view <div> Tip: To acquire your View Id, login to Google Analytics and navigate to your site data. Click the name of your site in the top menu bar. The identifiers for the available views are listed as numeric values in the right column. </div>
Connect String Options	The following is the default connect string option: <div> <code>Timeout=0;SupportEnhancedSQL=true;</code> </div> <ul style="list-style-type: none"> • The first option sets the connection timeout in seconds. Setting this value to 0 disables timeouts. • For more information on the second option, see below.
OAuth2 Client	The client is displayed. <div> NOTE: When you create the connection in this window, you must click Authenticate, which authenticates to the app. This step is required. </div>
Default Column Data Type Inference	Leave this value as Enabled .

For more information, see the driver documentation <http://cdn.cdata.com/help/DAG/jdbc/default.htm>.

Connect String Options

Enhanced SQL

By default, the following connect string options is included. This option enables an enhanced form of SQL support for the connection.

```
SupportEnhancedSQL=true;
```

When this feature is enabled:

- In-memory processing that is handled by default in the querying application (Designer Cloud application) is passed to Google Analytics for processing.
- This feature enables the use of advanced SQL expressions, such as the use of more predicates, joins, and aggregations.

For more information on enhanced SQL for Google Analytics, see https://cdn.cdata.com/help/DAG/jdbc/RSBGoogleAnalytics_p_SupportEnhancedSQL.htm.

Data Type Conversions

For more information, see the driver documentation <http://cdn.cdata.com/help/DAG/jdbc/>.

Example queries:

To import data from Google Analytics, you typically create a custom SQL SELECT statement. Example:

```
SELECT distinct Date, Sessions, NewUsers, BounceRate, PageviewsPerSession, AvgSessionDuration, Browser, FROM  
GoogleAnalytics.Traffic where StartDate = '90DaysAgo' and EndDate ='Today'
```

NOTE: The `distinct` keyword is currently required in default configuration when referencing specific values in a `where` clause. Optionally, you can enable the use of enhanced SQL for this connection, which eliminates the need for the `distinct` keyword. See "Connect String Options" above.

- Do not apply quotes, double-quotes, or brackets around field or table values.
- Literal values should be in single quotes.
- `StartDate` and `EndDate` parameters can be used to specify a date range.
 - You can also use date literal functions to specify date ranges. For more information, see http://cdn.cdata.com/help/DAG/jdbc/pg_dateliteralfunctions.htm.

For more information:

- SQL syntax: http://cdn.cdata.com/help/DAG/jdbc/pg_overview.htm
- SQL examples: http://cdn.cdata.com/help/DAG/jdbc/pg_retrievingdata.htm

DB2 Connections

Contents:

- *Prerequisites*
- *Configure*
 - *Create connection via API*
- *Reference*
 - *Connection URL*
 - *Driver Information*
- *Use*
- *Data Conversion*

You can create connections to one or more DB2 databases from Designer Cloud Powered by Trifacta® Enterprise Edition.

NOTE: This method of creating DB2 connections is supported for customer-managed installations of Designer Cloud Powered by Trifacta Enterprise Edition.

NOTE: Only connections to DB2 for Windows and Unix/Linux are supported.

Supported Versions: v10.5.5

Supported Environments:

Operation	Designer Cloud Powered by Trifacta Enterprise Edition	Amazon	Microsoft Azure
Read	Supported	Supported	Supported
Write	Not supported	Not supported	Not supported

Prerequisites

- If you haven't done so already, you must create and deploy an encryption key file for the Trifacta node to be shared by all relational connections. For more information, see *Create Encryption Key File*.

Configure

To create this connection:

- In the Import Data page, click the Plus sign. Then, select the Relational tab. Click the DB2 card.
- You can also create connections through the Connections page. See *Connections Page*.

Modify the following properties as needed:

Property	Description

Host	Enter your hostname. Example: <div>myDB2.example.com</div>
Port	Set this value to 50000.
Connect String Options	Please insert any connection options as a string here.
Database Name	Enter the name of the DB2 database to which to connect.
User Name	(basic credential type only) Username to use to connect to the database.
Password	(basic credential type only) Password associated with the above username.
Test Connection	After you have defined the connection credentials type, credentials, and connection string, you can validate those credentials.
Default Column Data Type Inference	Set to disabled to prevent the product from applying its own type inference to each column on import. The default value is enabled.
Connection Name	Display name of the connection
Connection Description	Description of the connection, which appears in the application.

Create connection via API

This connection can also be created using the API.

API: *API Reference*

- Type: jdbc
- Vendor: db2

For more information, see <https://api.trifacta.com/ee/es.t/index.html#operation/createConnection>

Reference

Connection URL

The properties that you provide are inserted into the following URL, which connects Designer Cloud Powered by Trifacta Enterprise Edition to the connection:

```
jdbc:trifacta:db2://<host>:<port>
```

Connect string options

The connect string options are optional. If you are passing additional properties and values to complete the connection, the connect string options must be structured in the following manner:

```
;<prop1>=<val1>;<prop2>=<val2>;...
```

where:

- <prop> : the name of the property
- <val> : the value for the property

delimiters:

- ; : any set of connect string options must begin and end with a semi-colon.
- = : property names and values must be separated with an equal sign (=).

Driver Information

This connection uses the following driver:

- **Driver name:** `com.trifacta.connect.jdbc.db2.DB2Driver`

NOTE: The driver in use is a proprietary version of the driver listed in the documentation. The behavior and property name are the same.

- **Driver version:** `DataDirect 5.1.4`
- **Driver documentation:**
<https://docs.progress.com/bundle/datadirect-connect-jdbc-51/page/DB2-Driver.html>

Use

For more information, see *Database Browser*.

Data Conversion

For more information on how values are converted during input and output with this database, see *DB2 Data Type Conversions*.

Job History

Through the Job History pages, you can review the status of your jobs in the Designer Cloud powered by Trifacta® platform , including downloading any available results.

Job History Page

Contents:

- *Flow Jobs*
- *Job Types*
- *Job Status*
- *Filter Jobs*

In the Job History page, you can track the status of all of your flow and sample jobs and plan runs.

- **Flow jobs:** These jobs execute transformations on your source data to generate outputs.
- **Sample jobs:** For more information on sampling jobs, see *Sample Jobs Page*.
- **Plan runs:** These jobs execute a sequence of tasks, which can include flow jobs. For more information, see *Plan Runs Page*.

Flow Jobs

You can only see jobs for the flows to which you have access in your current environment.

Flow jobs

Search jobs

StatusAll

Run by meAll jobs

Filter applied: Show all jobs run for the last 180 days

Job	Status	Flow	User	Started
<div>Categories</div> <div>Job ID: 3618594</div>	Completed	2021 POS		Last Sunday at 12:00 AM Ran for 2 minutes
<div>Categories</div> <div>Job ID: 3618514</div>	Completed	2021 POS		Last Saturday at 12:00 AM Ran for an hour
<div>Categories</div> <div>Job ID: 3618242</div>	Completed	2021 POS		09/25/2022 Ran for 42 minutes
<div>Categories</div> <div>Job ID: 3617870</div>	Completed	2021 POS		09/18/2022 Ran for 4 minutes
<div>Categories</div> <div>Job ID: 3617107</div>	Completed	2021 POS		09/11/2022 Ran for 2 minutes
<div>Categories</div> <div>Job ID: 3616377</div>	Failed	2021 POS		09/04/2022 Ran for 9 hours
<div>Categories</div> <div>Job ID: 3616086</div>	Failed	2021 POS		09/01/2022 Ran for 4 hours
<div>Categories</div> <div>Job ID: 3615884</div>	Failed	2021 POS		08/28/2022 Ran for 2 hours
<div>Categories</div> <div>Job ID: 3615544</div>	Failed	2021 POS		08/21/2022 Ran for a few seconds
<div>Categories</div> <div>Job ID: 3614857</div>	Failed	2021 POS		08/14/2022 Ran for 21 hours
<div>COVID-19_Daily_Testing_-_By_Person</div> <div>Job ID: 3614212</div>	Completed	SnowflakeExec		08/04/2022 Ran for a few seconds
<div>COVID-19_Daily_Testing_-_By_Person</div> <div>Job ID: 3614211</div>	Canceled	SnowflakeExec		08/04/2022 Ran for a few seconds

Figure: Job History page

Default view:

By default, the displayed jobs are:

- jobs that you have run. See "Job Tabs" below.
- for all job statuses. See "Job Status" below.
- from the last 180 days are displayed. You can change this filter as needed. See "Filter Jobs" below.

Job Tabs:

- **Run by me:** Jobs that you have initiated for the filtered time period are listed here.
- **All jobs:** All jobs that were initiated for assets that you own are listed here. For example, if a collaborator has run a job on a flow that you own, you can review its status here.

Columns:

- **Job:** Internal identifier for the job. This value is unique for all jobs in your Trifacta® instance.
 - Click the ID number to explore details about the job.
- **User:** The Trifacta user that initiated the job.
- **Run from:**
 - Location where the job was launched. Click the link to view details.
- **Status:**
 - See Tabs above.
- **Started:** Start timestamp for the job.
 - Scheduled jobs are indicated with an icon.

Actions:

- **Filter by status:** Click one of the tabs to filter the display to show only the listings for the selected job status.
- **Filter by type and date:** Click the Funnel icon to filter the list of jobs by source of execution, date range, or both. See below.
- **Search:** Enter text in the search field to filter the listed jobs by job ID, flow name, or dataset name.

Context menu:

Next to the job listing, click the options menu to see the following:

- **Cancel Job:** Select to cancel a job that is currently being executed.
- **Delete job:** Delete the job from the platform.

Deleting a job cannot be undone.

NOTE: This feature may not be enabled in your instance of the platform. For more information, please contact your Trifacta Administrator. See *Miscellaneous Configuration*.

- **Download Logs:** Download the logs for the job. If the job is in progress, log information is likely to be incomplete.

Tip: When jobs fail, the downloaded package includes additional configuration files and service logs to assist in debugging job execution issues. For more information, see *Support Bundle Contents*.

Additional options are available for each job. See *Job Details Page*.

Job Types

Each job listed in the Job History page is a grouping of related jobs acting on the same recipe and dataset(s). Each of these **jobgroups** breaks down into one or more of the following job types.

Tip: To review the status of individual jobs within a jobgroup, hover over the icons in the Status column for the jobgroup.

- **Pre-ingest SQL:** These jobs are SQL scripts that execute before the source data is ingested to the platform.
 - For additional details on these jobs, see the SQL scripts tab in the Job Details page. See *Job Details Page*.
 - For more information on these types of SQL scripts, see *Create Output SQL Scripts*.
- **Transform:** These jobs perform transformations on imported datasets based on the recipe from which the job was launched.
- **Profile:** If enabled as part of the job definition, a Profile job generates a visual summary of the results of your transformation job.
 - Profiling jobs may take longer than transformation jobs.
 - Even when selected, profiling jobs may not appear in the Job History page. In some cases, a profiling job may be folded into a transform job for optimization reasons.

NOTE: When the profiling job is run as part of the transform job, there is no listing for profiling in the mouse-over popup.

- See *Job Details Page*.
- **Publish:** Depending on multiple factors, your job may include a second Publish job that occurs after the Transform job. For example, job groups can include internal Publish jobs for writing results to the designated location in the base storage layer.

Publishing can also be executed as a separate, post-execution job. As needed, job results can be published from their target location to another location or data store. These jobs are tracked separately as Publish jobs and can be launched from the Job Details page.

- **Ingest:** For larger datasets from some relational connections, the platform transfers the data from the source to the default storage layer for faster processing. These ingest jobs occur before any transform or profiling takes place.
- **Post-ingest SQL:** These jobs are SQL scripts that execute after the job results have been published.
 - For additional details on these jobs, see the SQL scripts tab in the Job Details page. See *Job Details Page*.
 - For more information on these types of SQL scripts, see *Create Output SQL Scripts*.

Job Status

In the Status dropdown, each option corresponds to a possible status for jobs that have been initiated on the platform.

- **All jobs:** All jobs that you have initiated or were run on assets that you own.
- **Completed:** Job has successfully executed.

NOTE: Invalid steps in a recipe are skipped, and it's still possible for the job to be executed successfully.

NOTE: A warning icon may indicate that recipe errors were detected during the Transform phase. You can hover over the icon for more information.

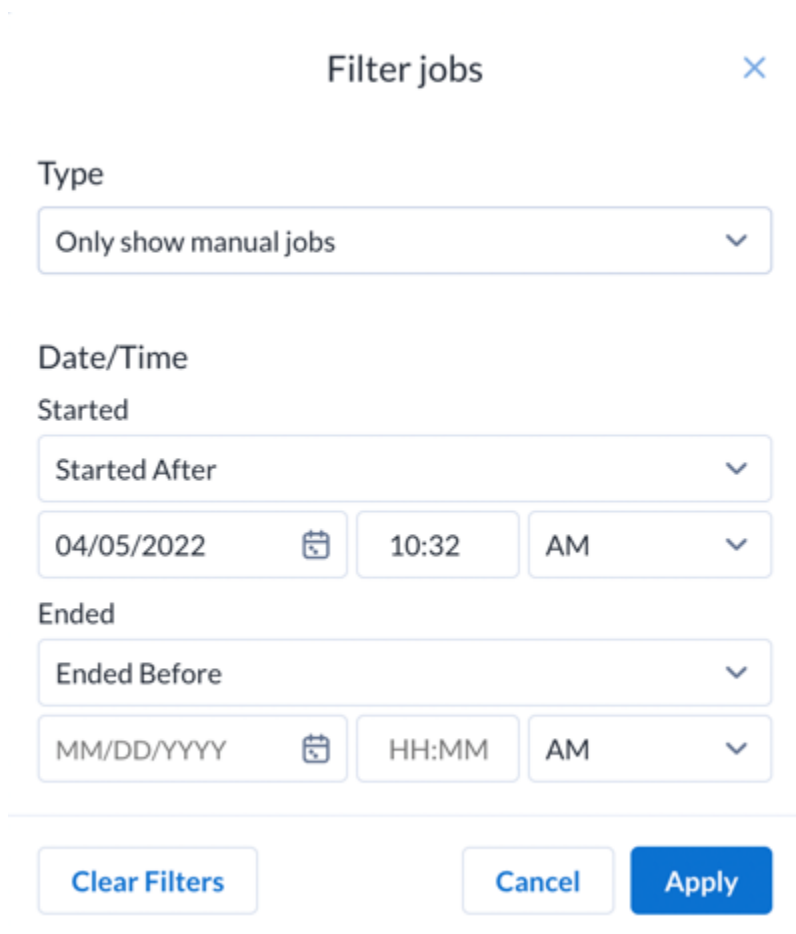
- **Failed:** job failed to complete.

NOTE: You can re-run a failed job from the Transformer page. If you have since modified the recipe, those changes are applied during the second run.

- **Publish Failed:** Some failed jobs may be listed under this status, which means that the publishing step of the configured job failed to complete.
- **Canceled:** Job was canceled.
- **Running:** Job is in progress.
- **Queued:** Job has been queued for execution.

Filter Jobs

To filter the list of jobs based on dates or source of execution, click the Funnel icon. You can use the following dialog to filter the display of jobs.



The image shows a 'Filter jobs' dialog box with a close button (X) in the top right corner. It contains three main sections: 'Type', 'Date/Time', and 'Ended'. The 'Type' section has a dropdown menu currently set to 'Only show manual jobs'. The 'Date/Time' section has a 'Started' dropdown set to 'Started After', followed by input fields for '04/05/2022' (with a calendar icon), '10:32', and 'AM' (with a dropdown arrow). The 'Ended' section has an 'Ended Before' dropdown, followed by input fields for 'MM/DD/YYYY' (with a calendar icon), 'HH:MM', and 'AM' (with a dropdown arrow). At the bottom, there are three buttons: 'Clear Filters', 'Cancel', and 'Apply'.

Figure: Filter Jobs dialog

Job type:

Show jobs based on the following available options:

- Show all jobs
- Only show manual jobs
- Only show scheduled jobs

Started:

- Specify the date and time when the jobs to display began.
- If needed, you can specify the start time as a range. Select `Start Between` from the drop-down list and populate both date-time rows.

Ended:

- Specify the date and time when the jobs to display ended.
- If needed, you can specify the end time as a range. Select `Ended Between` from the drop-down list and populate both date-time rows.

Actions:

- To clear the time period values, click **Clear Filters**.
- To apply the specified time filter to the Job History page, click **Apply**.

Change default filter:

An administrator can change the default number of days that are displayed. For more information, see *Workspace Settings Page*.

Job Details Page

Contents:

- *Overview Tab*
 - *Output Destinations Tab*
 - *Direct file download*
 - *Create imported dataset*
 - *Publish*
 - *SQL scripts Tab*
 - *Profile Tab*
 - *Dependency graph Tab*
 - *Data sources Tab*
 - *Parameters Tab*
 - *Webhooks Tab*
-

You can use the Job Details page to explore details about successful or failed jobs, including outputs, dependency graph, and other metadata. Download results to your local desktop or, if enabled, explore a visual profile of the data in the results for further iteration on your recipe.

NOTE: If the job is executed in an environment other than Trifacta Photon, the job is queued for execution in the environment. Jobs executed on a remote cluster may incur additional overhead to spin up execution nodes, which is typically within 10-15 seconds. During job execution, the Designer Cloud powered by Trifacta® platform observes the job in progress and reports progress as needed back into the application. The Designer Cloud powered by Trifacta platform does not control the execution of the job.

Page options:

- **Cancel job:** Click this button to cancel your job while it is still in progress. This button may not appear until the queued job has been submitted to the running environment.

NOTE: This option may not be available for all running environments. Job cancellation is not supported in high-availability deployments.

- **Publish results:** Publish your results to an external system. For more information, see *Publishing Dialog*.
- **Delete job:** Delete the job and its results.

Deleting a job cannot be undone.

NOTE: This feature may not be enabled in your environment. For more information, see *Miscellaneous Configuration*.

- **Download logs:** Download the log files associated with this job.

Tip: When jobs fail, the downloaded package includes additional configuration files and service logs to assist in debugging job execution issues. For more information, see *Support Bundle Contents*.

- **Download profile as PDF:** If visual profiling was enabled for the job, you can download the profile in PDF format.
- **Download profile as JSON:** If visual profiling was enabled for the job, you can download a JSON representation of the profile to your desktop.

Overview Tab

In the Overview tab, you can review the job status, its sources, and the details of the job run.

NOTE: If your job failed, you may be prompted with an error message indicating a job ID that differs from the listed one. This job ID refers to the sub-job that is part of the job listed in the Job summary.

Testflow > Untitled recipe - 2
Job 43
Finished Today at 11:29 AM

Publish results ...

Overview

Output destinations

Profile

Dependency graph

Data sources

Output data

#	POS_Sales	#	POS_Sales1	#	Store_Nbr	#	Item_Nbr	#	WM_Week	⌚	Daily
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	
2.67		2.67		92		427000		201049		2013/02/01	

17 columns 349954 rows This is a preview of the current data in your destination. It might not reflect the output from this particular job run.

Download View details

Job summary

Job ID 43

Job status Completed

Flow Testflow

Output Untitled recipe - 2

Execution summary

Job type Manual

User Administrator

Start time April 21st 2021, 11:29 am

Finish time April 21st 2021, 11:29 am

Last update April 21st 2021, 11:29 am

Duration a few seconds

Environment Photon

Optimization summary

Optimization Enabled

Completed stages

Transform with profile

Completed Today at 11:29 AM, started Today at 11:29 AM • Ran for 14 sec

Environment Photon

83% valid values 17% mismatching values 0.4% missing values

View steps and dependencies View profile

Publish

Completed Today at 11:29 AM, started Today at 11:29 AM • Ran for <1 sec

Activity

Untitled recipe _ 2.csv Completed • <1 sec

View all

Figure: Overview tab

You can review a snapshot of the results of your job.

Output Data:

The output data section displays a preview of the generated output of your job.

NOTE: This section is not displayed if the job fails.

You can also perform the following:

- **View:** If it is present, you can click the View link to view the job results in the datastore where they were written.

NOTE: The View link may not be available for all jobs.

- **Download :** If it is present, click the Download link to download the generated job results to your local desktop.
- **View details:** Click **View details** to view the generated results in the side bar. See the Output Destinations below.

Completed Stages:

This panel provides information on the progress and completion status of each stage of the job execution.

Tip: Depending on the operation, you may be able to monitor transfer rate performance for larger datasets.

- **Connect:** The platform is attempting to connect to the datastore hosting the asset sources for the datasets.
- **Schema validation:** When enabled, the schemas of a job's datasources are checked as the first step of job execution.
 - Datasets with changes in them are reported at the top of the list. Click **View all** to see schema validation for all of the datasets used in the job in the Data sources tab.
 - Optionally, the job can be halted if there are differences between the schema that is read and the schema that has been stored from the previous job run. This option can prevent data corruption.
 - If no errors are detected, then the job is completed as normal.
 - For more information on schema validation, see *Overview of Schema Management*.
- **Request:** The platform is requesting the set of assets to deliver.
- **Ingest:** Depending on the type of source data, some jobs ingest data to the base storage layer in a converted format before processing begins. This ingested data is purged after job completion.
- **Prepare:** (Publishing only) Depending on the destination, the Prepare phase includes the creation of temporary tables, generation of manifest files, and the fetching of extra connections for parallel data transfer.
- **Transfer:** Assets are transferred to the target, which can be the platform or to the output datastore.
- **Transform:** This stage covers the execution of your recipe steps in order to transform the source data.
- **Profile:** If you chose to profile your output data, this stage is completed after transformation is complete. Results are available in the Profile tab.

NOTE: If you chose to generate a profile of your job results, the transformation and profiling tasks may be combined into a single task, depending on your environment. If they are combined and profiling fails, any publishing tasks defined in the job are not launched. You may be able to ad-hoc publish the generated results. See below.

- **Publish:** This stage covers the writing of the outputs of the transformed data. These outputs are available through the Output destinations tab.
- **Process:** Cleanup after data transfer, including the dropping of temporary tables or copying data within the instance.

For more information, see *Overview of Job Monitoring*.

If present, you can click the **Show Warnings** link to see any warnings pertaining to recipe errors, including the relevant step number. To review the recipe and dependencies in your job, click **View steps and dependencies**. See the Dependencies tab below.

- If you chose to profile results of your job, click **View profile** to review. See Profile tab below.
 - A visual profile provides a graphical snapshot of the results of a successful transformation job for the entire dataset and individual columns in the dataset.
- If your job output specified SQL scripts to run before or after job execution, you can track their progress in the following stages:
 - **Pre-ingest SQL:** Script that is configured to run before the source data is ingested to the platform.
 - **Post-publish SQL:** Script that is configured to run after the output data has been published.
 - For additional details, see the SQL scripts tab below.
 - For more information on SQL scripts in job execution, see *Create Output SQL Scripts*.

Publish:

You can also review the outputs generated as a result of your job. To review and export any of the generated results, click **View all**. See Outputs Destinations tab below.

Job summary:

- **Job ID:** Unique identifier for the job

Tip: If you are using the REST APIs, this value can be used to retrieve and modify specifics related to this job.

- **Job status:** Current status of the job:
 - **Queued:** Job has been queued for execution.
 - **Running:** Job is in progress.
 - **Completed:** Job has successfully executed.

NOTE: Invalid steps in a recipe are skipped, and it's still possible for the job to be executed successfully.

- **Failed:** Job failed to complete.

NOTE: You can re-run a failed job from the Transformer page. If you have since modified the recipe, those changes are applied during the second run.

- **Canceled:** Job was canceled.

- **Flow:** Name of the flow from which the job was executed. Click the link to open the flow.
- **Output:** Name of the output object that was used to define the generated results. Click the link to open the output.

Execution summary:

- **Job type:** The method by which the job was executed:
 - **Manual** - Job was executed through the application interface.
 - **Scheduled** - Job was executed according to a predefined schedule. See *Add Schedule Dialog*.

User: The user who launched the job

Environment: Where applicable, the running environment where the job was executed is displayed.

Start time: Timestamp for when processing began on the job. This value may not correspond to when the job was queued for execution.

Finish time: Timestamp for when processing ended on the job, successful or not

Last update: Timestamp for when the job was last updated

Duration: Elapsed time of job execution

Optimization summary:

For jobs sourced from relational datasets, you can optionally enable SQL-based optimizations, which apply some of the steps specified in your recipe back in the datasource, where they can be executed before the data is transferred to the running environment for execution. Using these optimizations means faster performance based on a lower volume of data transfer.

Workspace administrators must enable the optimization feature for the workspace. For more information, see *Workspace Settings Page*.

When the feature is enabled, optimizations must be enabled for each flow. You can also select the optimizations to apply.

When optimizations have been applied to your flow, they are listed on the Overview tab:

Optimization: This setting is displayed if flow optimizations have been enabled for this flow.

Columns pruned: If one or more unused columns have been pruned in the datasource via SQL, the count of columns is listed here.

Filters pushed down: If one or more row filters has been applied in the datasource via SQL, the count of filters is listed here.

If an optimization is disabled or was not applied to the job run, it is not listed.

Output Destinations Tab

If the job has successfully completed, you can review the set of generated outputs and export results.

test_flow > REF_PROD - 3

Job 60

Finished Today at 4:16 PM

Publish results

...

Overview

Output destinations

Profile

Dependency graph

Data sources

You can download the generated results locally or [publish](#) to another storage location.

Name	Location	Status
REF_PROD _ 3.csv	hdfs://hadoop:8020/trifacta/queryResults/admin@trifacta.local/REF_PROD _ 3.csv	Completed • 13 sec

Figure: Output Destinations tab

Actions:

For each output, you can do the following:

View details: View details about the generated output in the side bar.

Tip: The View details panel contains breakdowns for each phase of a job. If the job fails, you can review error messages, which correspond to entries in the Data Service log file.

Download result: Download the generated output to your local desktop.

NOTE: Some file formats may not be downloadable to your desktop. See below.

Create imported dataset: Use the generated output to create a new imported dataset for use. See below.

NOTE: This option is not available for all file formats.

Direct file download

Click one of the provided links to download the file through your browser to your local desktop.

NOTE: If these options are not available, data download may have been disabled by an administrator.

HYPER: You can download HYPER formatted outputs to your desktop.

If you have generated output in a Tableau format and have configured a connection to Tableau Server, you can publish directly to the server. See *Publishing Dialog*.

Create imported dataset

Optionally, you can turn your generated results into new datasets for immediate use in the Designer Cloud powered by Trifacta® platform. For the generated output, select **Create imported dataset** from its context menu.

NOTE: If you generated results in Parquet format only, you cannot create a dataset from it, even if the Create button is present. This is a known issue.

NOTE: When you create a new dataset from your job results, the file or files that were written to the designated output location are used as the source. Depending on your backend datastore permissions are configured, this location may not be accessible to other users.

Publish

If the Designer Cloud powered by Trifacta platform is connected to an external storage system, you may publish your job results to it. Requirements:

Your version of the product supports publishing.

Your connection to the storage system includes write permissions.

Your results are generated in a format that the target system supports for writing.

All sub-jobs, including profiling, successfully completed.

For more information, see *Publishing Dialog*.

SQL scripts Tab

If the output for your job included one or more pre- or post-job SQL script executions, you can review the status of their execution during the job.

NOTE: If a SQL script fails to execute, all downstream phases of the job fail to execute.

Tip: If the SQL script execution for this job encountered errors, you can review those errors through this tab. For more detailed information, click **Download logs**.



2013 POS > Untitled recipe Job 19 Finished Today at 5:48 PM				Publish results	...
Overview	Output destinations	SQL scripts	Dependency graph	Data sources	
Connection	SQL statement	Settings	Status		
postgres	CREATE TABLE IF NOT EXISTS "public"."spotable" (timestamp date, jobType varchar(255), jobStatus varchar(255)); INSERT INTO "public"."spotable"(timestamp, jobType, jobStatus) VALUES ('2021-06-22','Trifacta','started');	Run before data ingest	Completed • 3 sec		
postgres	CREATE TABLE IF NOT EXISTS "public"."spotable" (timestamp date, jobType varchar(255), jobStatus varchar(255)); INSERT INTO "public"."spotable"(timestamp, jobType, jobStatus) VALUES ('2021-06-22','Trifacta','completed');	Run after data publish	Completed • <1 sec		

Figure: SQL scripts tab

Columns:

Connection: Name of the connection through which the script was executed.

SQL statement: The first part of the SQL script that was executed.

Settings:

- Run before data ingest - script was executed pre-job.
- Run after data publish - script was executed post-job, after the job results had been written.

Status: Current status and execution duration of the SQL script.

NOTE: If you have multiple SQL scripts for each settings, they may execute in parallel. For example, if you created three pre-job SQL scripts, there is no guarantee that they executed in the order in which they are listed.

View details:

Hover over a SQL script entry and click **View details**.

In the SQL script details window, you can review:

Connection and SQL of the executed script.

Any error messages that occurred during execution.

Tip: To review log information for any error messages, click **Download logs**.

For more information on these types of SQL scripts, see *Create Output SQL Scripts*.

Profile Tab

Review the visual profile of your generated results in the Profile tab. Visual profiling can assist in identifying issues in your dataset that require further attention, including outlier values.

NOTE: This tab appears only if you selected to profile results in your job definition.

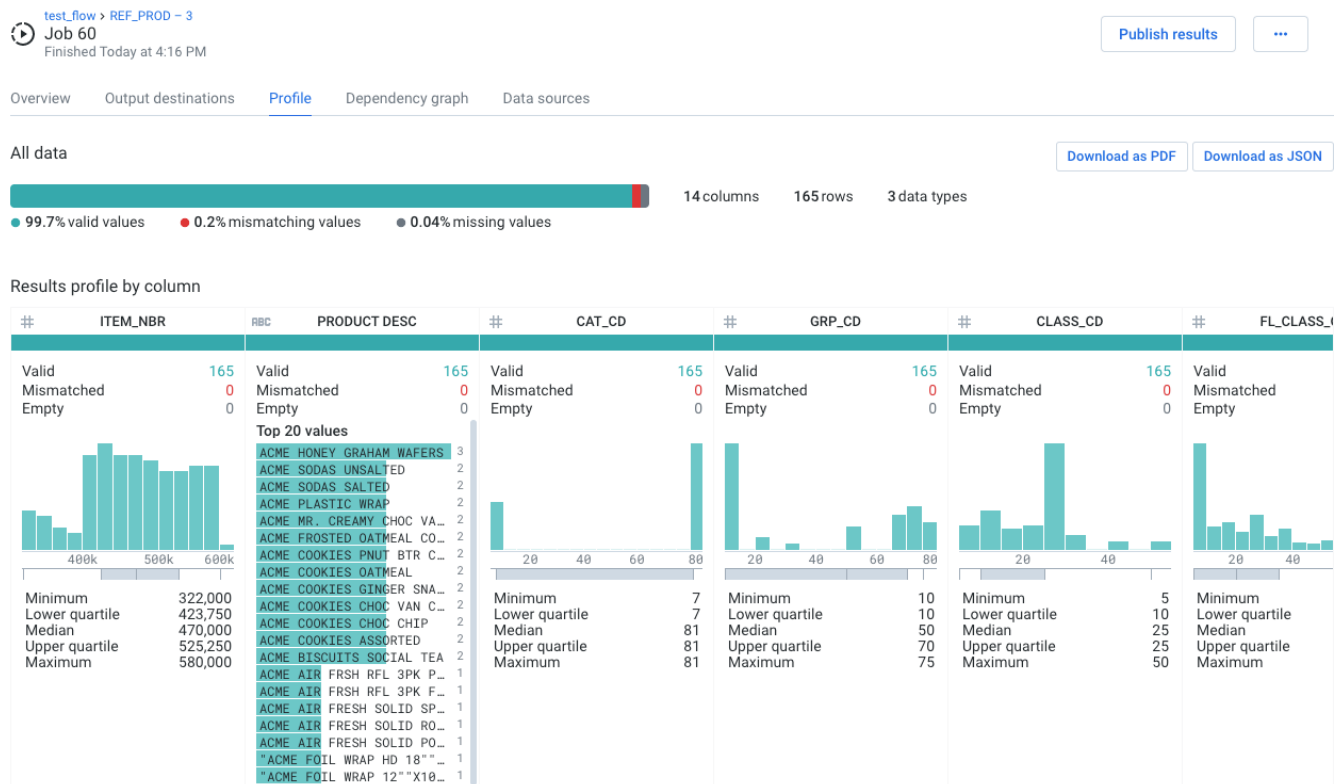


Figure: Profile tab

Download as PDF: Download your visual profile and results of your data quality rules on the entire dataset as a PDF file.

Download as JSON: Download your visual profile as a JSON file.

In particular, you should pay attention to the mismatched values and missing values counts, which identify the approximate percentage of affected values across the entire dataset.

NOTE: The computational cost of generating exact visual profiling measurements on large datasets in interactive visual profiles severely impacts performance. As a result, visual profiles across an entire dataset represent statistically significant approximations.

NOTE: Designer Cloud Powered by Trifacta Enterprise Edition treats null values as missing values. Imported values that are null are generated as missing values in job results (represented in the gray bar).

Tip: Mouse over the color bars to see counts of values in the category.

Tip: Use the horizontal scroll bar to see profiles of all columns in wide datasets.

In the lower section, you can explore details of the transformations of individual columns. Use this area to explore mismatched or missing data elements in individual columns.

Depending on the data type of the column, varying information is displayed.

Tip: You should review the type information for each column, which is indicated by the icon to the left of the column.

Dependency graph Tab

In this tab, you can review a simplified representation of the flow from which the job was executed. This flow view displays only the recipes and datasets that contributed to the generated results.

Tip: To open the full flow, you can click its name in the upper-left corner.

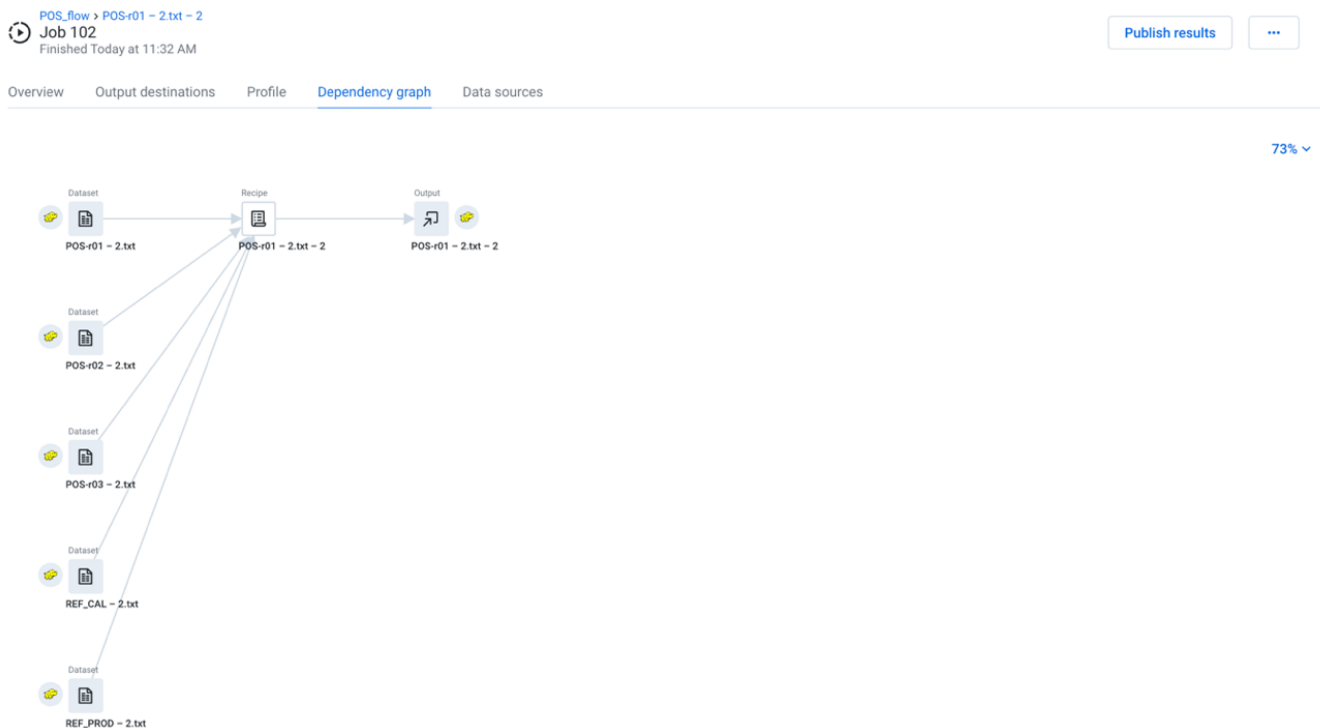


Figure: Dependency graph tab

Zoom menu:

You can zoom the dependency graph canvas to display areas of interest in the flow graph.

The zoom control options are available at the top-right corner of the dependency graph canvas. The following are the available zoom options:

Tip: You can use the keyboard shortcuts listed in the zoom options menu to make quick adjustments to the zoom level.

Zoom in: Zoom in 10% on the canvas to focus on greater detail.

Zoom out: Zoom out 10% from the canvas to see more of it.

Zoom to fit: Change the zoom level to fit all of the objects of your flow onto the screen.

25%, 50%, or 100%: Change the zoom level to one of the preset levels.

Recipe actions:

Download recipe: Download the text of the recipe in Wrangle .

Display Wrangle /natural language: Toggle display of the recipe in raw language or in readable language.

Limitations:

- You can select only recipes in the flow graph.
- Context controls and menus are not available.

Data sources Tab

In the Data sources tab, you can review all of the sources of data for the executing recipe.

Schema_drift > Untitled recipe
Job 2705
Finished Last Friday at 3:46 AM












Overview	Output destinations	Profile	Dependency graph	Data sources
Name	Location	Schema validation	Ingestion	
 unload_all_data_types_del17	public/unload_all_data_types_del17	✔ No schema changes found	✔ Completed • 50 sec	
 unload_all_data_types_del3	public/unload_all_data_types_del3	✔ No schema changes found	✘ Canceled	
 unload_all_data_types_del10	public/unload_all_data_types_del10	✔ No schema changes found	✘ Canceled	
 unload_all_data_types_del1	public/unload_all_data_types_del1	✔ No schema changes found	✔ Completed • 47 sec	
 unload_all_data_types_del14	public/unload_all_data_types_del14	✔ No schema changes found	⚠ Completed • 2 min	
 unload_all_data_types_del16	public/unload_all_data_types_del16	✔ No schema changes found	✘ Canceled	
 unload_all_data_types_del18	public/unload_all_data_types_del18	✔ No schema changes found	✘ Canceled	
 unload_all_data_types_del19	public/unload_all_data_types_del19	✔ No schema changes found	✔ Completed • 57 sec	
 unload_all_data_types_del11	public/unload_all_data_types_del11	✔ No schema changes found	✘ Canceled	
 unload_all_data_types_del15	public/unload_all_data_types_del15	✔ No schema changes found	✔ Completed • 56 sec	
 unload_all_data_types_del13	public/unload_all_data_types_del13	✔ No schema changes found	⚠ Completed • 2 min	

Figure: Data sources tab

NOTE: If a flow is unshared with you, you cannot see or access the datasources for any jobs that you have already run on the flow, including any PDF profiles that you generated. You can still access the job results. This is a known issue.

Schema validation:

If schema validation has been enabled, you can review validation errors for individual datasets. For more information, see *Schema Changes Dialog*.

Datasets with parameters:

If your source is a dataset with parameters, you can review and count the individual files that were matched and imported.

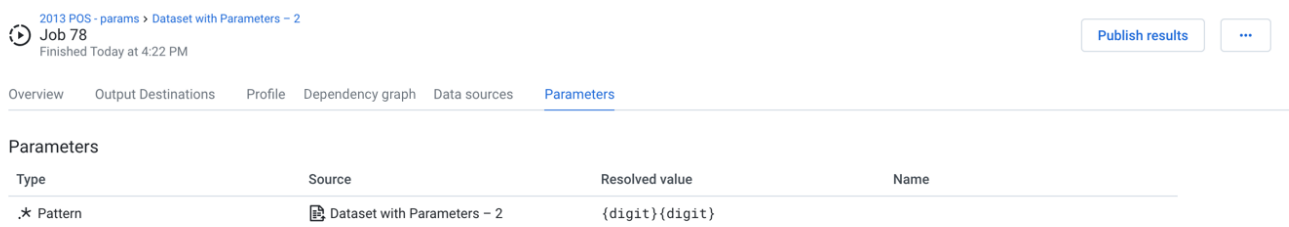
For the imported dataset, click **View details**. Then, click the Files tab in the context panel.

This tab can be a good check to ensure that you have specified your dataset parameters correctly.

Parameters Tab

If your flow references parameters, you can review the state of the parameters at the time of job execution.

NOTE: This tab appears only if the job is sourced from a flow that references parameters. For more information, see *Overview of Parameterization*.



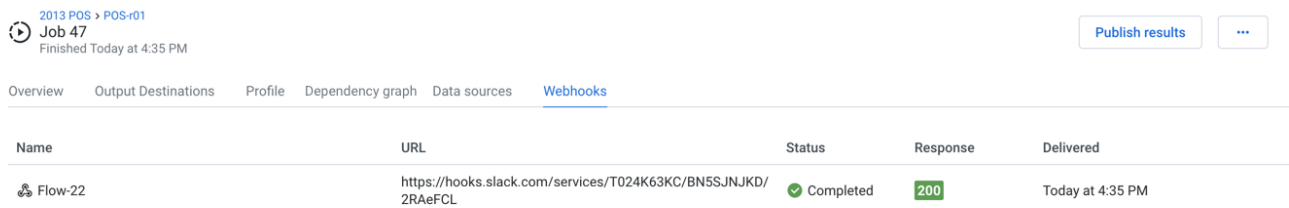
The screenshot shows the 'Parameters' tab for a job named 'Job 78' (2013 POS - params > Dataset with Parameters ~ 2). The job finished today at 4:22 PM. The 'Parameters' section contains a table with the following data:

Type	Source	Resolved value	Name
* Pattern	Dataset with Parameters ~ 2	{digit}{digit}	

Figure: Parameters tab

Webhooks Tab

Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.



The screenshot shows the 'Webhooks' tab for a job named 'Job 47' (2013 POS > POS-r01). The job finished today at 4:35 PM. The 'Webhooks' section contains a table with the following data:

Name	URL	Status	Response	Delivered
Flow-22	https://hooks.slack.com/services/T024K63KC/BN5SJNJKD/2RAeFCL	Completed	200	Today at 4:35 PM

Figure: Webhooks Tab

When a webhook task has been triggered for this job, you can review the status of its delivery to the target system.

Webhooks are defined on a per-flow basis. For more information, see *Create Flow Webhook Task*.

NOTE: Webhook notifications may need to be enabled in your environment. See *Workspace Settings Page*.

Columns:

Name: Display name for the webhook task.

URL: Target URL where the webhook notification is delivered.

Status: HTTP status code returned from the delivery of the message.

- 200 - message was delivered successfully.

Delivered: Timestamp for when the webhook was delivered.

Publishing Dialog

Contents:

- *Limitations*
- *Publish to Tableau Server*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When a job has successfully completed, you can publish your job results to one of your connected datastores. In the Job Details page, click the Output Destinations tab. Then, click **Publish**.

Limitations

- You cannot publish ad-hoc results for a job when another publishing job is in progress for the same job through the application. Please wait until the previous job has been published before retrying to publish the failing job. This is a known issue.

JSON-formatted files that are generated by the Designer Cloud powered by Trifacta® platform are rendered in JSON Lines format, which is a single-line per-record variant of JSON. For more information, see <http://jsonlines.org>.

Publish

Hive

Database

default

Table

DatasetsARRAYINDEXOF1_trifacta

Data Option

Create new table & load data

Publish

Publish to Tableau Server



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

If you have created a Tableau Server connection, you can export results that have been generated to the connected server.

Supported Formats:

- **Hyper:** Results are written to your Tableau Server in Hyper format.

Options:

- **Connection:** If you have created multiple connections to Tableau Server, please select the connection to use from the list.
 - The Site name is specified as part of the connection. See *Tableau Server Connections*.
- **Project Name:** Name of the Tableau Server project.
- **Datasource Name:** Name of the Tableau Server datasource. This value is displayed for selection in Tableau Server.

Data Option:

If you are publishing to a pre-existing table, schema validation is automatically performed.

- **Create new datasource:** The platform creates the datasource and then loads it with the results from this job. If you attempt to use this option on a source that already exists, the publishing job fails, and an error is generated in the log.
- **Append data to existing datasource:** The results from this job are appended to the data that is already stored in Tableau Server. If you attempt to append to a source that does not exist, the publishing job fails, and an error is generated in the log. Append operations also fail if you publish to a target with a different schema.
- **Replace contents of existing datasource:** Target datasource is dropped. A new datasource is created using the schema of the generated output and filled with the job results.

Troubleshooting - Request timeout exception

When publishing to Tableau Server, you may encounter an error similar to the following for a PUT operation in the job log:


```
com.trifacta.clients.http.exceptions.RequestTimeoutException: PUT request to ...
```

Solution:

In this case, the size of individual chunks submitted to Tableau Server is too large. The PUT operation did not complete before a server timeout was encountered, and the operation failed.

To address this issue, you should lower the size of each chunk that is submitted to Tableau Server for publication. For more information, see *Configure Data Service*.

Schema Changes Dialog

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When schema validation is enabled, the schemas of the job's datasets are checked against stored versions of the schema from the previous job run and reported in the Job Details page. Through the Schema Changes dialog, you can explore differences (findings) between the stored and read schemas for each dataset.

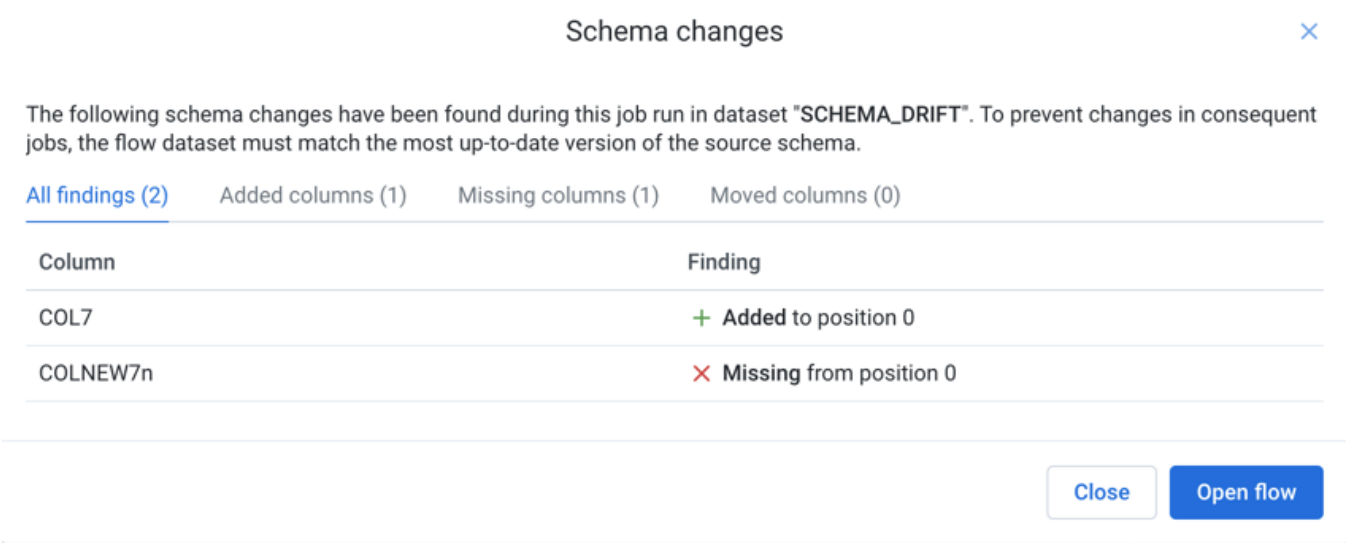


Figure: Schema Changes Dialog

Columns:

- **Column:** Name of the column in the dataset.
- **Finding:** Description on the change between the column in the stored schema and the column in the schema read during this job execution. Additional details on these finds is below.

Actions:

Use the tabs at the top of the screen to filter the list of findings.

Tip: In all cases, these issues should be explored.

NOTE: Renamed columns appear as a deletion of the original column and an addition of the renamed column.

Finding Type	Description	Risk
Added column	The indicated number of columns have been added to the dataset.	Risk: Moderate If these columns are added at the end of the dataset, they may not cause breakages. However, they may be omitted from some dataset reshaping transformation steps. If these columns are inserted in the middle of the dataset, they have caused some columns to be moved. See below.

Missing column	The indicated number of columns are missing in the new dataset.	Risk: High Depending on the use of the missing columns, these could cause errors in your job execution. Additionally, downstream consumers of your data may be counting on these columns.
Moved column	The indicated number of columns have been moved to new locations in the dataset.	Risk: Moderate Named references to the columns should still function. However, if you reference column ranges to specify a selection of columns, then these steps may no longer include the moved columns.


- Use the controls at the bottom to move to additional pages of findings.
- To open the flow from which the job and these findings were detected, click **Open flow**.
- To close the dialog, click **Close**.

Limits:

Maximum number of findings per type:

- Columns added: 125
- Columns missing: 125
- Columns moved: 50
- Total number of findings: 300

Plan Runs Page

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

In the Plan Runs page, you can track the status of all runs of your plans.

You can only see runs for the plans to which you have access in your current environment.

Plan runs are executed from the Plans page. See *Plans Page*.

Plan runs

Run by meAll runs

StatusAll

Plan run	Status	User	Started
<div><div>abc78440] Test Slack Task</div><div>Run ID: 456411</div><div>Post a message</div><div><div>POST</div><div>https://slack.com/api/chat.postMessage</div></div></div>	<div>Completed</div>	Steve Olson (you)	03/13/2022 <div>Ran for 8 hours</div>
<div><div>Run POS 2021</div><div><div>POS 2021</div><div>Job ID: 3604116</div></div><div>Post a message</div><div><div>POST</div><div>https://slack.com/api/chat.postMessage</div></div></div>	<div>Completed</div>		03/13/2022 <div>Ran for a few seconds</div>
<div><div>Post a message</div><div><div>POST</div><div>https://slack.com/api/chat.postMessage</div></div></div>	<div>Skipped</div>		
<div><div>abc78440] Test Slack Task</div><div>Run ID: 456359</div></div>	<div>Completed</div>		03/06/2022 <div>Ran for a day</div>
<div><div>abc78440] Test Slack Task</div><div>Run ID: 456341</div></div>	<div>Completed</div>		02/27/2022 <div>Ran for 12 hours</div>
<div><div>abc78440] Test Slack Task</div><div>Run ID: 456337</div></div>	<div>Completed</div>		02/20/2022 <div>Ran for 8 hours</div>
<div><div>abc78440] Test Slack Task</div><div>Run ID: 456282</div></div>	<div>Failed</div>		02/13/2022 <div>Ran for 2 hours</div>

Figure: Plan Runs page

Default view:

By default, the displayed runs are:

- runs that you have executed. See "Plan Run Tabs" below.
- for all run statuses. See "Plan Run Status" below.
- from the last 180 days are displayed. You can change this filter as needed. See "Filter Plan Runs" below.

Plan Run Tabs:

- **Run by me:** Plan runs that you have initiated for the filtered time period are listed here.
- **All runs:** All plan runs that were initiated for assets that you own are listed here. For example, if a collaborator has run a plan that you own, you can review its status here.

Columns:

- **Run:** Internal identifier for the plan run. This value is unique for all runs in your Trifacta® instance.
 - Click the ID number to explore details about the plan run, including execution of individual tasks in the plan. See *Plan View Page*.
- **Status:** The current status of the run. See "Plan Run Status" below.

- **User:** The Trifacta user that initiated the run.
- **Started:** Start timestamp for the run.

Actions:

- **Filter by status:** Select an option in the Status dropdown to filter the display to show only the listings for the selected status.
- **Filter by type and date:** Click the Funnel icon to filter the list of plans by source of execution, date range, or both. See below.
- **Search:** Enter text in the search field to filter the listed runs by run ID, user name, or status name.

Plan Run Status

In the Status dropdown, each option corresponds to a possible status for plan runs that have been initiated on the platform.

- **All runs:** All runs that you have initiated are listed here.
- **Completed:** Run has successfully executed.
- **Failed:** Run failed to complete.

NOTE: You can re-run a failed task by clicking the **Retry from failed** button. For more information, see *Plan Run Details Page*.

To download the logs of a failed plan run, select **Download logs** from the run's context menu.

- **Canceled:** Run was canceled.
- **Running:** Run is in progress.

Filter Plan Runs

To filter the list of plans, click the Funnel icon next to the search bar. You can use the following dialog to filter the display of plans based on dates or source of execution:

Filter plan runs

×

Type

Show all plan runs

Date/Time

Started

Started After

01/01/2021

📅

11:30AM

▼

Ended

Ended Before

MM/DD/YYYY

📅

HH:MMAM

▼

Clear Filters

Cancel

Apply

Figure: Filter plan runs dialog

Plan type:

Show plans based on the following available options:

- **Show all plan runs**
- **Only show manual plan runs**
- **Only show scheduled plan runs**
- **Only show api plan runs**

Started:

- Specify the date and time when the plan runs to display started.
- Select **Started Between** or **Started After** from the drop-down list and populate both date-time rows.

Ended:

- Specify the date and time when the plan runs to display ended.
- Select **Ended Between** or **Ended After** from the drop-down list and populate both date-time rows.

Actions:

- To apply the specified time filter to the Plan Runs page, click **Apply**.
- To clear the time period values, click **Clear Filters**.

Change default filter:


Copyright © 2022 Trifacta Inc.


Page #366

An administrator can change the default number of days that are displayed. For more information, see *Workspace Settings Page*.

Sample Jobs Page

In the Sample Jobs page, you can track the status of all sample jobs to which you have access.

 Sample jobs



[Run by me](#) [All jobs](#)

Status

All













Job	Status	Flow	User	Started
 Random Job ID: 3614502	 Completed	 Collab COLLABORATORS		08/08/2022 Ran for 4 minutes
 Random Job ID: 3469759	 Completed	 2021 POS 2021 POS		02/14/2022 Ran for 3 minutes
 Random Job ID: 3469750	 Failed	 2021 POS 2021 POS		02/14/2022 Ran for 2 minutes
 Random Job ID: 3447779	 Completed	 2013 POS Untitled recipe - 2		02/10/2022 Ran for a minute

Figure: Sample jobs page

Default view:

By default, the displayed jobs are:

- jobs that you have run. See "Job Tabs" below.
- for all job statuses. See "Job Status" below.
- from the last 180 days are displayed. You can change this filter as needed. See "Filter Sample Jobs" below.

Job Tabs:

- **Run by me:** Jobs that you have initiated for the filtered time period are listed here.
- **All jobs:** All jobs that were initiated for assets that you own are listed here.

Columns:

- **Jobs:** Sample jobs to which you have access.

Tip: Click the link for the sample type to load the sample.

- **Status:** The current status of the sample jobs.
- **Flow:** Name of the flow in which the samples have been collected. You can click the flow link to go the Flow View Page.
- **User:** The Trifacta user who initiated the sample job.
- **Started:** Start timestamp for the sample job.

Actions:

- **Filter by status:** Select an option from the Status dropdown to filter the display to show only the listings for the selected status.
- **Filter by type and date:** Click the Funnel icon to filter the list of sample jobs by date range, time of execution, or both. For more information, see below.
- **Search:** Enter text in the search field to filter the listed sample jobs by job ID, sample type, status, or flow.

Tip: Select **Download logs** from the sample job's context menu to download the log file for that sample job.

Sample Job Status

In the Status dropdown, each option corresponds to a possible status for Sample jobs that have been initiated in the Designer Cloud application .

- **All jobs:** All sample jobs that have started.
- **Completed:** Sample jobs that have successfully completed.
- **Failed:** Sample jobs that have failed.
- **Canceled:** Sample jobs that were canceled.

Tip: To cancel an in-progress sample job, select the **Cancel job** option while the sample job is running.

- **Running:** Sample jobs that are currently in progress.
- **Queued:** Job has been queued for execution.

Filter Sample Jobs

To filter the list of sample jobs, click the Funnel icon next to the search bar. Use the dialog to filter the display of samples based on date ranges:

Filter sample jobs

Date/Time

Started

Started After

01/01/202210:59AM

Ended

Ended Before

MM/DD/YYYYHH:MMAM

Clear Filters

Cancel

Apply

Figure: Sample Job dialog

Started:

- Specify the date and time when the sample jobs have started.
- Select **Started Between** or **Started After** from the drop-down list and populate both date-time rows.

Ended:

- Specify the date and time when the sample jobs have ended.
- Select **Ended Between** or **Ended Before** from the drop-down list and populate both date-time rows.

Actions:

- To clear the time period values, click **Clear Filters**.

Change default filter:

An administrator can change the default number of days that are displayed. For more information, see *Workspace Settings Page*.

Diagnose Failed Jobs

Contents:

- [Job Types](#)
- [Identify Job Failures](#)
 - [Invalid file paths](#)
 - [Jobs that Hang](#)
 - [Spark Job Error Messages](#)
 - [Databricks Job Errors](#)
- [Try Other Job Options](#)
- [Review Logs](#)
 - [Hadoop logs](#)
- [Learn More](#)

Use these guidelines and features to begin the process of diagnosing jobs that have failed.

Job Types

The following types of jobs can be executed in the Designer Cloud powered by Trifacta® platform :

- **Conversion jobs:** Some datasources, such as binary file or JSON formats, must be converted to a format that can be easily read by the Designer Cloud application . During data ingestion, the datasource is converted to a natively supported file format and stored on backend storage.
- **Transform job:** This type of job executes the steps in your recipe against the dataset to generate results in the specified format. When you configure your job, any set of selected output formats causes a transform job to execute according to the job settings.
- **Profile job:** This type of job builds a visual profile of the generated results. When you configure your job, select **Profile Results** to generate a profile job.
- **Publish job:** This job publishes results generated by the platform to a different location or datastore.
- **Ingest job:** This job manages the import of data from a JDBC source into the default datastore for purposes of running a transform or sampling job.

Tip: Information on failed plan executions is contained in the `orchestration-service.log` file, which can be acquired in the support bundle. For more information, see [Support Bundle Contents](#).

NOTE: For each collected sample, a sample job ID is generated. In the Samples panel, you can view the sample job IDs for your samples. These job IDs enable you to identify the sample jobs in the Sample Jobs page.

Identify Job Failures

When a job fails to execute, a failure message appears in the Job History page:

35	Automated Demo	airports_info Airport Flow	Transform Finished Today at 10:25 AM Environment: Hadoop	Completed	Today at 10:26 AM Ran for a minute
33	Automated Demo	airports_info Airport Flow	Profile Finished Today at 10:25 AM	Failed	Today at 10:24 AM Ran for 2 minutes
32	elmer	Customer_D Getting Star	Publish Failed Today at 10:25 AM 1 Failed	Completed	Today at 10:24 AM Ran for a few seconds

Figure: Publish job failed

In the above example, the Transform and Profile jobs completed, but the Publish job failed. In this case, the results exist and, if the source of the problem is diagnosed, they can be published separately. From the job's context menu, select **Download Logs**. You can download the jobs logs to look for reasons for the failure. See below.

Invalid file paths

When your job uses files as inputs or outputs, you may receive invalid file path errors. Depending on the backend datastore, these can be one of the following:

- Path to the file is invalid for the current user. Path may have been created by another user that had access to the location.
- Path contains invalid characters in it. For more information, see *Supported File Formats*.
- Resource was deleted.

Jobs that Hang

In some cases, a job may stay in a pending state indefinitely. Typically, these errors are related to a failure of the job tracking service. You can try the following:

- Resubmit the job.
- Have an administrator restart the platform. See *Start and Stop the Platform*.
- Submit the job again.

Spark Job Error Messages

The following error messages may appear in the Designer Cloud application when a Spark job fails to execute.

"Aggregate too many columns" error

Your job could not be completed due to one or more Pivot, Window or other Aggregation recipe steps having too many aggregate functions in the `Values` parameter.

Solution: Please split these aggregates across multiple Aggregation steps.

"Binary sort" error

Sorting a nested column such as an array or map is not supported.

Codegen error

Your job could not be completed due to the complexity of your recipe.

Tips:

- Look to break up your recipe into sequences of recipes. You can chain recipes together one after another in Flow View.
- If you have complex, multi-dataset operations, you should try to isolate these into smaller recipes.
- Use sampling to checkpoint execution after complex steps.

"Colon in path" error

Your job references one or more invalid file paths. File and folder names cannot contain the colon character.

"Invalid input path" error

Your job references one or more invalid file paths. File names cannot begin with characters like dot or underscore.

"Invalid union" error

Union operations can only be performed on tables with compatible column types.

Tip: Edit the union in question. Verify that the columns are properly aligned and have consistent data types. For more information, see *Union Page*.

"Job service unreachable" error

There was an error communicating with the Spark Job Service.

Tip: An administrator can review the contents of the `spark-job-service.log` file for details. See *System Services and Logs*.

"Oom" error

When you encounter out of memory errors related to job execution, you should review the following general items related to your flow.

General Tips:

- Review your recipes to see if you can identify ways to break them up into smaller recipes.
- Operations such as joins and unions can greatly increase the size of your datasets.
- Resource consumption is also affected by the complexity of your recipe(s).
- If you suspect that there are several jobs running in parallel, you can drop the job launch batch size to 2 or 1, which serializes job execution while preserving memory. For more information, see *Configure Application Limits*.
- You might be able to configure overrides to the Spark settings to allocate more memory for job execution.
 - This feature may need to be enabled in your environment. See *Enable Spark Job Overrides*.
 - See *Configure User-Specific Props for Cluster Jobs*.

"Path not found during execution" error

One or more datasources referenced by your job no longer exist.

Tip: Review your flow and all of its upstream dependencies to locate the broken datasource. Reference errors for upstream dependencies may be visible in downstream recipes.

"Too many columns" error

Your job could not be completed due to one or more datasets containing a large number of columns.

Tip: A general rule of thumb is to avoid over 1000 columns in your dataset. Depending on your environment, you may experience performance issues and job failures on narrower datasets.

"Version mismatch" error

The version of Spark installed on your Hadoop cluster does not match the version of Spark that the Designer Cloud powered by Trifacta platform is configured to use.

Tip: For more information on the appropriate version to configure for the product, see *Configure for Spark*.

Databricks Job Errors

The following error messages are specific to Spark errors encountered when running jobs on Databricks.

NOTE: When a Databricks job fails, the failure is immediately reported in the Designer Cloud application . Collection of the job log files from Databricks occurs afterward in the background.

Tip: A platform administrator may be able to download additional logs for help in diagnosing job errors.

"Runtime cluster" error

There was an error running your job.

"Staging cluster" error

There was an error launching your job.

Try Other Job Options

You can try to re-execute the job using different options.

Tips:

- **Disable flow optimizations.** If your job is using data from a relational source that supports pushdowns, you can try to disable flow optimizations and then re-run the job. For more information, see *Flow Optimization Settings Dialog*.
- **Look to cut data volume.** Some job failures occur due to high data volumes. For jobs that execute across a large dataset, you can re-examine your data to remove unneeded rows and columns of data. Use the Deduplicate transformation to remove duplicate rows.
- **Gather a new sample.** In some cases, jobs can fail when run at scale because the sample displayed in the Transformer page did not include problematic data. If you have modified the number of rows or columns in your dataset, you can generate a new sample, which might illuminate the problematic data. However, gathering a new sample may fail as well, which can indicate a broader problem.
- **Change the running environment.** If the job failed on Trifacta Photon, try executing it on another running environment.

Tip: The Trifacta Photon running environment is not suitable for jobs on large datasets. Use the default running environment to execute your job.

Review Logs

Job logs

In the listing for the job on the Job History page, click **Download Logs** to send the job-related logs to your local desktop.

NOTE: If encryption has been enabled for log downloads, you must be an administrator to see a clear-text version of the jobs listed below. For more information, see *Configure Support Bundling*.

When you unzip the ZIP file, you should see a numbered folder with the internal identifier for your job on it. If you executed a transform job and a profile job, the ZIP contains two numbered folders with the lower number representing the transform job.

`job.log`. Review this log file for information on how the job was handled by the application.

Tip: Search this log file for `error`.

Support bundle: If support bundling has been enabled in your environment, the `support-bundle` folder contains a set of configuration and log files that can be useful for debugging job failures.

Tip: Please include this bundle with any request for assistance to *Alteryx Support*.

For more information on configuring the support bundle, see *Configure Support Bundling*.

For more information on the bundle contents, see *Support Bundle Contents*.

Support logs

For support use, the most meaningful logs and configuration files can be downloaded from the application. Select **Resources menu > Download logs**.

NOTE: If you are submitting an issue to *Alteryx Support*, please download these files through the application.

For more information, see *Download Logs Dialog*.

The admin version of this dialog enables downloading logs by timeframe, job ID, or session ID. For more information, see *Admin Download Logs Dialog*.

Trifacta node logs

NOTE: You must be an administrator to access these logs. These logs are included when an administrator downloads logs for a failed job. See above.

On the Trifacta node, these logs are located in the following directory:

```
<install_dir>/logs
```

This directory contains the following logs:

- `batch-job-runner.log`. This log contains vital information about the state of any launched jobs.

- `webapp.log`. This log monitors interactions with the web application.

Issues related to jobs running locally on the Trifacta Photon running environment can appear here.

Hadoop logs

In addition to these logs, you can also use the Hadoop job logs to troubleshoot job failures.

- You can find the Hadoop job logs at port 50070 or 50030 on the node where the ResourceManager is installed.
- The Hadoop job logs contain important information about any Hadoop-specific errors that may have occurred at a lower level than the Designer Cloud application , such as JDK issues or container launch failures.

Contact Support

If you are unable to diagnose your job failure, please contact *Alteryx Support*.

Learn More

- <https://community.trifacta.com/s/article/how-to-use-the-trifacta-job-log-files>
- <https://community.trifacta.com/s/article/troubleshooting-a-hadoop-job-failure>

Schedules

Contents:

- *Overview*
 - *Interface*
 - *Tasks*
 - *Topics*
-

Schedules enable you to execute jobs on your assets in the Designer Cloud powered by Trifacta® platform .

Overview

For more information on scheduling in the platform, see *Overview of Scheduling*.

Interface

To review the set of available schedules, click **Schedules**. See *Schedules Page*.

- To modify a schedule, select the link for it in the Schedules page.
- To create a new schedule, click **New** in the Schedules page.
- See *Add Schedule Dialog*.

Tasks

For more information, see *Schedule a Job*.

Topics

Overview of Scheduling



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Contents:

- *Limitations*
 - *Data Management*
 - *Flows for scheduling*
 - *Schedule a Job*
 - *Job Execution*
 - *Tracking*
 - *Configure*
-

As needed, you can schedule the execution of recipes in your flows on a recurring basis. For example, if the source file of your flow is updated outside of the application on a weekly basis, you can define a schedule to execute the recipe associated with the related imported dataset after the data has been refreshed. When the scheduled job successfully executes, you can collect the wrangled output in the specified output location, where it is available in the published form that you have specified.

To schedule a job, you must create the following configuration objects:

1. **Define a schedule** - For each flow you can define a schedule. A **schedule** specifies one or more recurring times (**triggers**) when scheduled jobs for the flow are executed. For example, in a single schedule, you can specify daily trigger times for incremental updates and monthly execution times for rollups.

Tip: The scheduler supports a modified form of cron job syntax. For more information, see *cron Schedule Syntax Reference*.

2. **Define one or more scheduled destinations** - When you specify a **scheduled destination** for a recipe, the recipe is executed whenever one of the schedule's execution times occurs. Scheduled destinations are specified like regular destinations.

NOTE: When a schedule for a flow is triggered, all of recipes to generate the scheduled destinations are executed. Manual destinations are not generated. You cannot create schedules for individual outputs.

Limitations

- One schedule cannot be applied to multiple flows.
- You cannot create separate schedules for individual recipes within a flow. A schedule defined at the flow level applies to all recipes in the flow.
- Only a flow owner can create or modify a flow's schedule.

Data Management

NOTE: Since scheduled destinations are re-populated with each scheduled execution, you must determine how you wish to manage the data that is published to each location. Data management should be done outside of the Designer Cloud powered by Trifacta® platform .

- **Import:** Before each scheduled execution, you should refresh the source of the imported dataset with new data outside of the Designer Cloud powered by Trifacta platform .
- **Execution:** Please verify that the publishing settings for your scheduled destination are consistent with how you are using the results. For example, if the scheduled destination creates a new file with the same name for each execution (replace), you must move the generated file out of the output location before the next scheduled execution.
- **Output:** You must collect the generated results. While you can export the job's results through the Job History page, you may find it easier to use an external scheduler to gather the results and forward to the downstream consumer of them.

Flows for scheduling

Tip: When a schedule is executed, all outputs in a flow are generated, even if they are unused. For better performance on larger flows, you can create a separate flow that contains only the references back to the objects in the source flow that you wish to have scheduled. As an additional benefit, this separation keeps development and scheduled execution in separate flows.

Schedule a Job

Schedules and scheduled destinations are defined through the Schedules page. See *Schedules Page*.

Job Execution

Tracking

You can monitor a scheduled job like any other job in the application. See *Job History Page*.

Configure

See *Configure Scheduling*.

Schedules Page

In the Schedules page, you can review the available set of schedules for the Designer Cloud powered by Trifacta® platform .



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

NOTE: This page is available to project owners and workspace administrators only.

A **schedule** is the automated execution of an output in a flow once or on a recurring basis. Schedules are composed of the following:

- A schedule
- A set of one or more scheduled destinations
- These objects are created from the flow. For more information, see *Flow View Page*.
 - See *Add Schedule Dialog*.

NOTE: Schedules owned by users that have been disabled continue to execute. An admin can disable them through the Schedules page.

Schedules				
Owner	Frequency	Flow	State	Last Updated
Administrator (you)	Weekly: 12:00 AM - On Sunday	census data	Enabled	Today at 11:16 AM
Administrator (you)	Cron: 0 0 ** TUE * - At 12:00 AM, only on...	2013 POS	Enabled	Today at 11:14 AM

Figure: Schedules page

Columns:

- **Owner:** The user that owns the schedule.
- **Frequency:** The frequency of occurrence of the schedule.
 - **Cron:** Cron jobs utilize a modified form of cron scheduling syntax to define execution time. For more information, see *cron Schedule Syntax Reference*.
 - **Weekly/Monthly/Daily:** You can also schedule jobs to execute according to a regular calendar period.
- **Flow:** Name of the flow to which the schedule applies.
 - If available, you can click the link to open the flow.
- **State:** The current state of the schedule:
 - **Enabled** - The schedule is active and will execute at the next occurrence according to the frequency.
 - **Disabled** - The schedule is inactive and cannot be executed until it is enabled.
- **Last Updated:** Timestamp for the when the schedule was last modified.

Actions:

- **Filter by update:** Click the caret next to Last Updated to sort the list of schedules.
- **Search:** Enter text in the search field to filter the listed jobs by flow name.

Context menu:

Next to the schedule listing, click the options menu to see the following:

- **Enable/Disable Schedule:** Select this option to toggle availability of the schedule.
- **Delete Schedule:** Delete the schedule.

Deleting a schedule is permanent and cannot be undone.

Add Schedule Dialog

As needed, you can create a new schedule to execute jobs.

To add a schedule to your flow, click the drop-down menu in the Flow View page, and select **Schedule**.



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

NOTE: Flow owners can create schedules. Other users must have the appropriate privileges to create schedules for the flow.

NOTE: By default, when scheduled or API jobs are executed, no validations are performed of any writesettings objects for file-based outputs. Issues with these objects may cause failures during transformation or publishing stages of job execution. Jobs of these types should be tested through the Designer Cloud application first. A workspace administrator can disable the skipping of these validations.

NOTE: For a dataset with parameters, scheduled times are used when resolving date range parameters.

NOTE: Do not schedule executions through Flow View in a Prod instance when Deployment Manager is enabled.

- Schedules defined in Flow View are applied to Active and Non-Active releases in Production environments.
- If the scheduled release is deactivated, the schedule still exists, and the jobs are executed on an flow that is now out-of-date.
- For more information, see *Overview of Deployment Manager*.

Add Schedule
×

Scheduling Options

Timezone

America/Los_Angeles
▼

Frequency

Weekly
▼

on

Sunday
▼

at

12:00
AM
▼

Add

Variables

Set variable values for this schedule's executions

< > regionNum

01

Cancel

Save

Figure: Add Schedule dialog

Scheduling Options

- **Timezone:** Select the timezone to apply to the schedule.
 - To use UTC time zone, select `UTC` in the drop-down. For more information, see *Supported Time Zone Values*.
- **Frequency:**
 - **Hourly, Daily, Weekly, Monthly:** Run the schedule at the specified moment for the interval.

Tip: For drop-downs showing days of the week or days of the month, you can click multiple values to select them.

- **cron:** Set the schedule according to cron syntax.
 - Time zone settings set in the drop-down are used with the cron schedule.
 - For more information, see *cron Schedule Syntax Reference*.

To add another trigger for the flow's schedule, click **Add**.

To create the schedule, click **Save**.

Parameter Overrides

If your flow contains one or more variable parameters, you can apply overrides to any variables. When the scheduled job is executed, the variable value is applied to job at runtime.

NOTE: In a trigger, displayed parameter values may be inherited from the plan or flow that is being triggered. To ensure that the proper value is used, you should set a specific value for the override in the trigger. This is a known issue.


For each listed variable, you can modify the value that is applied when the schedule is executed.

For more information, see *Overview of Parameterization*.

Schedule a Job

Contents:

- *Add a Schedule*
- *Schedule a Destination*
- *Edit Schedule*
- *Disable Schedule*
- *Delete*
 - *Delete a schedule*
 - *Delete a destination*

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can define scheduled executions of your jobs to deliver outputs to known locations. You can schedule execution of jobs on source data, which can be replenished with fresh data asynchronously.

NOTE: Before you begin, you should verify that your data management pipeline into and out of the platform has been appropriately defined. This pipeline includes how data is written to the output location. For more information, see *Overview of Scheduling*.

When you schedule a job, you create two objects:

NOTE: You must create both of these objects to schedule a job execution.

Object	Description
Schedule	A schedule applies to the entire asset. It contains one or more intervals at which jobs are executed to deliver outputs include a scheduled destination.
Scheduled destination	A scheduled destination is an output location, format, and other settings that is populated with the results of execution.

Add a Schedule

Steps:

1. Click **Schedules** in the Designer Cloud application .
2. From the context menu, select **Schedule**.
3. In the Add Schedule dialog, select your scheduling options:
 - a. Timezone: Select the timezone to use to determine when to execute the specified schedule.
 - b. Frequency: Select the time and frequency of execution: Hourly, Daily, Weekly, Monthly, or cron.

NOTE: Scheduling supports a modified version of cron scheduling syntax. For more information, see *cron Schedule Syntax Reference*.

- c. To add another scheduled time, click **Add**.
4. To save your schedule, click **Save**.

A Calendar icon appears in Flow View to indicate that the flow has a schedule associated with it.

Schedule a Destination

Steps:

1. To specify a destination for your schedule, click the recipe you wish to execute at the scheduled time.
2. If you have not done so already, click the Output icon next to the recipe to create an output for it.
3. In the right panel, locate the Scheduled destinations header. Click **Add**.
4. Specify an output location, format, and updating method.
5. Click **Save**.

Edit Schedule

- To edit the scheduled times, click the Calendar icon. Then, click **Edit**. Make changes as needed and save.
- To edit a scheduled destination, select the output in Flow View. In the right panel, click **Edit** next to the appropriate scheduled destination.

Disable Schedule

- To disable a schedule you control, locate the schedule in the Schedules page. From the schedule's context menu, click **Disable**.
- Administrators can disable any schedule in the Schedules page.

Delete

Delete a schedule

Locate the schedule in the Schedules page. From the schedule's context menu, click **Delete**.

Delete a destination

Tip: If you have deleted the schedule for the flow, you do not need to delete the scheduled destination. It cannot run without a schedule.

1. In Flow View, select the output.
2. In the right panel, select **Delete Output** from the context menu.

Schedule a Plan

After you have created a plan, you can schedule regular execution of the plan run.

Steps:

1. In Plan View, click **Schedule**.
2. In the right panel, click **Create schedule**.
3. In the right panel, you can specify the trigger, which initiates the execution of your plan.
 - a. Specify the time zone and the frequency at which the plan is executed.

Tip: The **Designer Cloud powered by Trifacta® platform** supports a modified form of cron scheduling syntax.

- b. If your tasks include parameters, you can apply overrides to these parameters when the flows are executed as part of the plan.
4. To save your plan trigger, click **Save**.

The plan run executes at the scheduled time.

Profile

Through your profile, you can review and update your account information, connection settings, API access tokens, and more.

Select the **Profile** menu to explore.

Preferences Page

Contents:

- *User Preferences*
 - *Profile*
 - *Account*
 - *Email notifications*
 - *Sessions*
 - *Storage*
 - *Databricks settings*
-

From the Preferences page, you can manage aspects of your user account and other settings. Select **User menu** > **Preferences**.

User Preferences

Profile

Review and manage your user profile. For more information, see *User Profile Page*.

Account

Change your password and set your locale among other preferences. See *Account Settings Page*.

Email notifications

Configure your personal preferences on receiving email notifications about activities in the product.

NOTE: This feature requires access to an SMTP server to send emails. For more information, see *Enable SMTP Email Server Integration*.

NOTE: This feature may need to be enabled in your workspace. See *Workspace Settings Page*.

For more information, see *Email Notifications Page*.

Sessions

Review the list of devices that have logged into your account. You can revoke any unrecognized and unauthorized sessions. For more information, see *Sessions Page*.

Storage

Review and modify the locations where you store data.

See *Storage Config Page*.

Databricks settings

Each user must save a Databricks Personal Access Token to their user account. See *Databricks Settings Page*.

User Profile Page

In your user profile, you can review your personal information and update your photo. Select **User menu > Preferences**.

NOTE: After saving changes to your user profile and exiting, please refresh the page.

Profile

Photo



[Upload photo](#)

Email

Name

Figure: User Profile Page

Profile settings

Name: Display name for your Trifacta® account.

Email: Email address associated with your account

NOTE: This value is the user ID. It must be a valid email address and cannot be modified after registration.

Upload photo

NOTE: This feature may need to be enabled. See *Miscellaneous Configuration*.

You can upload a preferred image associated with your user account. This image appears wherever the application contains a personal identifier, such as the icon for the User menu.

Image requirements:

- Format: JPG (JPEG), PNG, GIF, SVG, BMP, WEBP
- Dimensions: square dimensions work best. If you are using a non-square image, you should center the image details along the shorter edge of the image.

Steps:

1. Below the icon at the top of the User Profile page, click **Upload photo**.
2. Navigate your local desktop.
3. Select the file and click **Open**.
4. The icon is replaced by the image from the file you uploaded.

Storage Config Page

The Storage Config page allows you to configure storage locations for where you upload datasets and generate results.

Tip: When editing a directory location, click the Pencil icon. You can paste URLs to storage locations into the textbox.

The following options are available for configuring your storage environment.

NOTE: If you cannot modify the values in this page, you may need to enable the feature to modify user paths. For more information, see *Workspace Settings Page*.

Output home directory

Relative path to the directory where your results are stored by default and where your samples are stored. Click **Edit** to modify the home directory where results are stored.

Full path concatenates Output Protocol/Host value and this value.

Do not modify this value unless directed to do so. This path is not validated. If you specify a path to a directory to which you do not have appropriate permissions, all job exports will fail.

NOTE: Multiple users cannot share the same home directory.

NOTE: If your HDFS environment is encrypted, any location that you specify to write results for a job must be in the same encryption zone as this directory. For more information, please contact your HDFS administrator.

Upload directory

Relative path to the directory where your uploads are stored. To modify this value, click **Edit**.

NOTE: This setting only applies if the Designer Cloud powered by Trifacta® platform is connected to a backend datastore.

NOTE: You cannot upload to locations to which you do not have write access.

AWS credentials

If per-user mode is enabled, this option allows workspace members to apply individual key-secret values and roles to their accounts and modify other personal storage settings. For more information, see *Configure Your Access to S3*.

Account Settings Page

Contents:

- *Change Password*
 - *Locale*
 - *Other Settings*
-

In your Account Settings page, you can change your locale and modify other settings related to your account.

NOTE: After saving changes to your account settings and exiting, please refresh the page.

Change Password

NOTE: Users can reset their own passwords if a workspace administrator has enabled self service password reset. See *Workspace Settings Page*.

NOTE: If Single Sign-On (SSO) has been enabled, then these options are not available.

Tip: You can modify your password directly through the `/change-password` URL.

Old password: To change your password, enter your current password here.

New password: To change your password, enter a new password here.

To save the password, click **Update password**.

Tip: Forgot your password? Click **Reset password via email** to send a reset password to the email address registered for your account.

Locale

Select the locale to use when validating data types in the application.

NOTE: After saving changes to your locale, refresh your page. Subsequent executions of the data inference service use the new locale settings.

NOTE: When locale is changed, data type validation is affected only on subsequent executions of data type inference. If you are using structured datasets, such as schematized files, data types may be attached to the datasets that you have already imported. These data types are not affected.

For more information, see *Locale Settings*.

Other Settings

Enable keyboard shortcuts: When enabled, you can use keyboard shortcuts in the workspace.

Tip: When keyboard shortcuts are enabled, press ? in the application to see the available shortcuts.

Share usage data to improve product intelligence: When collaborative suggestions are enabled, anonymized data on how you use the product is aggregated with other workspace users' data to improve the suggestions provided by the product to all workspace users. You can use this setting to opt-out of sharing your data.

NOTE: This data is not shared with Alteryx or other users.

NOTE: If this setting is not present, the feature is disabled in your workspace. For more information, see *Workspace Settings Page*.

Email Notifications Page

Contents:

- *Receive emails about flow jobs*
- *Receive emails about plan runs*
- *Receive emails when a flow or plan is shared with you*
- *Delivery address*
- *Timezone*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

You can configure the notifications that are sent to your email address and other related settings.

Notifications can be sent for jobs or plans where you are the owner or a collaborator.

NOTE: Email notifications requires that you configure the Designer Cloud powered by Trifacta platform to use an available SMTP server. For more information, see *Enable SMTP Email Server Integration*.

NOTE: Email notifications may need to be enabled in your environment. For more information, see *Workspace Settings Page*.

Tip: If visual profiling has been enabled, a PDF version of the profile of the job results is included as an attachment to the email.

Receive emails about flow jobs

When enabled, you can receive email notifications about job activity. You can receive emails from:

- flows where you are an owner or collaborator
- flows where someone has added you as a watcher

Tip: The email provides a summary of Data Quality rules (rules that were run and the success and failures of those rules) and the profile details (valid, mismatched, and missing) when a job is completed successfully.

Email notifications are configured on a per-flow basis.

Receive emails about plan runs

When enabled, you can receive email notifications about plan runs if you are an owner or collaborator.

Email notifications are configured on a per-plan basis. For more information, see *Manage Plan Notifications Dialog*.

Receive emails when a flow or plan is shared with you





Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When enabled, you automatically receive notifications to your registered email address when a flow or plan is shared with you.

Delivery address

If needed, you can send notification emails to a different email address.

Uses:

- Your login email address cannot receive email.
- You wish to deliver email to an alias within your enterprise.
- You wish to deliver your email to a third-party application, such as Slack.

Tip: You can configure the sender email address and sender name that is used for all emails generated by the Trifacta node. For more information, see *Enable SMTP Email Server Integration*.

Timezone

Select the timezone in which email dates are displayed.

Sessions Page

The Sessions page enables you to view the number of devices signed into your account. You can use this information to make sure no one else has signed in to your account. You can review the devices that are authorized and revoke any unfamiliar devices.

Sessions

These are the devices that have logged into your account. You can revoke any sessions that you don't recognize.


Device	IP address	Last activity	First log in	
Chrome on Mac	76. [REDACTED]	Current session	Nov 15, 2021, 9:14 AM	
Chrome on Mac	76. [REDACTED]	Nov 15th 2021, 9:10 AM	Nov 15, 2021, 9:10 AM	

Figure: Sessions page


Columns:

- **Device:** Name of the device that is connected to your account.
- **IP address:** Unique address that identifies the device on the internet or a local network.
- **Last activity :** Displays the recent activity you performed on the device.
- **First log in:** Timestamp details at which the first session was logged.

Removing unrecognized devices:

Click the Delete icon next to any unrecognized device.

Access Tokens Page

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

From the Access Token page, you can generate and manage access tokens that apply to your account. Access tokens can be used when accessing the REST APIs.

Tip: Access token usage is the preferred method of authenticating from API. See <https://api.trifacta.com/ee/es.t/index.html#section/Authentication>

NOTE: If this page is not visible, the feature has not been enabled in your instance of the platform. See *Enable API Access Tokens*.

NOTE: Workspace administrators can choose to enable creation and use of access tokens to individual workspace users. For more information, see *Workspace Settings Page*.

NOTE: Workspace administrators can delete the access tokens of other users.

Access Tokens

Generate New Token


Token ID	Description	Status	Created	Expires	Last used
0bc1d49f-5475-4c62-a0ba-6ad269389ada	new token	 Active	2019-01-15 12:58:28	2020-01-15 12:58:28	Never Used

Figure: Access Tokens Page

Actions:

- **Generate New Token:** Click to generate a new access token for your user account. See below.

Columns:

- **Token ID:** Internal identifier for the token

NOTE: This is not the token itself. That value is exposed when you create the token and must be retained for any use outside of the Designer Cloud application .

- **Description:** User-provided description of the token
- **Status:** Current status of the token:
 - Active - Token is active and can be used for access.

- **Expired** - Token has expired after its expiration timestamp has been reached.
- **Created**: Timestamp for when the token was created.
- **Expires**: Timestamp for when the token expires.
- **Last Used**: Timestamp for when the token was last used.

Context menu:

- **Delete Token**: Click the delete the token.

Deleting a token cannot be undone.

NOTE: If you delete an active token, any API usage that references the token no longer works.

Generate Token

When you generate a token, you can provide the following information.

Generate Token

×

Lifetime (days)

365

Set lifetime to -1 to have the token never expire.

Description (optional)

Cancel

Generate

Figure: Generate Token Dialog

- **Lifetime**: Enter the number of days that you would like to use this token without renewal.
 - If the token expires, a new one must be created. You can generate a new token at any time.
 - You can generate any number of tokens.

Tip: Depending on your environment, you may be able to set this value to -1 to never expire the token.

- **Description**: (Optional) you can provide a user-friendly description of the purpose of the token. This value is for display purposes only.

To create the token, click **Generate**.

After the token is generated, it is automatically activated. You can have multiple active tokens.

You must copy the token out of the application. Click **Copy Token to clipboard** to copy the text value of the token.

For security purposes, after you close the My Token screen, the token is no longer accessible. You cannot reopen this dialog. You must copy this value and store it in a secure place for later use.

Databricks Settings Page

To access a Databricks cluster for running jobs, each user of the Designer Cloud powered by Trifacta® platform must insert their personal access token into their profile. This configuration enables the user to authenticate to a connected Databricks cluster.

NOTE: Each user must insert a personal access token into their profile. Users that do not provide a personal authentication token cannot run jobs on Databricks, including transformation, sampling, and profiling jobs.

NOTE: When you reset your personal access token (PAT), a new cluster is created if your new token does not have access to your current cluster. If you are resetting an expired personal access token, no new cluster is created. This new cluster is created when you first request access to the Databricks cluster. When you next use an interface that require access to the cluster, such as the relational browser, it may take some time to load.

Prerequisites

- Acquire your Databricks personal access token.

NOTE: Your Databricks personal access token must be acquired from the same region as your Databricks deployment. This region name is available through the Designer Cloud application .

For more information, see <https://docs.azure.databricks.net/api/latest/authentication.html#requirements>.

Steps

1. Login to the application. From the menu bar, select **User menu > Preferences > Databricks**.
2. **Configure URL:**
 - a. **For Azure developments:** Acquire the Azure Databricks personal access token from the same region as your Azure Databricks deployment. The region name is available through the Designer Cloud application . For more information, see *Configure for Azure Databricks*.
 - b. **For AWS developments:** Edit the workspace URL, as required and click **Save**.
 - The existing property `databricks.serviceUrl` is used to configure the URL to the Databricks Service to run Spark jobs. For more information, see *Configure for AWS Databricks*.
 - The `databricks.serviceUrl` defines the default Databricks workspace for all user in the Designer Cloud Powered by Trifacta Enterprise Edition workspace.
 - You can override the default settings in this page.
3. **Personal access token:** In the Personal Access Token field, paste your token.
 - a. To use a different token, click **Change**.
4. **Databricks table cluster name:** Each user can specify the name of a cluster to use to browse a Databricks Tables deployment.

NOTE: This cluster must be created and maintained by your Databricks administrator. This cluster can be shared among multiple users.

5. **Databricks policy name:** To select the cluster policy to use when you are executing jobs on the cluster, click **Edit**. The available policies are listed in the drop-down.

NOTE: Cluster policies determine characteristics of Databricks clusters that are used or created for job execution. This feature requires additional configuration.

- a. For more information, see *Configure for AWS Databricks*.
 - b. For more information, see *Configure for Azure Databricks*.
6. Click **Save**.

Account Management Tasks

The topics below provide information on how to manage aspects of your account in the Designer Cloud powered by Trifacta® platform .

Change Password

To recover your password, click **Forgot password?** in the login screen.

To change your password after you have logged in, select your name from the User menu. Enter a new password, confirm it, and save your changes. The new password is applied when you next try to log in. See *User Profile Page*.

The password for the administrator account should be changed immediately after installation is complete. See *Change Admin Password*.

Configure Your Access to S3

Contents:

- *Getting Started*
 - *Credential Provider*
 - *IAM Role*
 - *AWS Key and Secret*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

If per-user access to S3 has been enabled in your Trifacta® deployment, you can apply your personal S3 access credentials through the AWS Credentials page. You can use the following properties to define the S3 buckets to use for uploads, job results, and temporary files.

Getting Started

You can access these settings through the Designer Cloud application .

Steps:

1. In the menu bar, click the User menu.
2. Select **Storage**. click **Edit** for AWS Credentials and Storage Settings, where you can review and modify your S3 access credentials.

Credential Provider

IAM Role

NOTE: This role must be created through AWS for you. For more information, please contact your AWS administrator.

Tip: This method is recommended for access AWS resources.

AWS Storage Settings

✕

Please see [AWS Config Settings](#) for help completing this form.

Credential Provider

✓
IAM Role

AWS Key and Secret

Available IAM Role ARNs

Select Default IAM Role ARN

Default S3 Bucket

This bucket will contain uploaded files, temporary files, and job results.

Cancel

Save

Figure: Apply your IAM role and credentials

Setting	Description
Available IAM Role ARNs	You can specify the set of IAM Role ARN to use to authenticate to AWS resources.
Select Default IAM Role ARN	From the available IAM Role ARNs, you can specify the default one.
Default S3 Bucket	This bucket is used for storage, unless another bucket is explicitly selected. <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>NOTE: Specify the top-level bucket name only. There should not be any backslashes in your entry.</p> </div>
Extra S3 Buckets	You can specify a comma-separated string of additional S3 buckets that are available for storage. Do not put any quotes around the string. Whitespace between string values is ignored.

AWS Key and Secret

Per-user access must be enabled by your Trifacta administrator. See [S3 Access](#).

AWS Storage Settings

×

Please see [AWS Config Settings](#) for help completing this form.

Credential Provider

IAM Role

✓

AWS Key and Secret

AWS Access Key

AWS Secret Key

Default S3 Bucket

3fac-testing

This bucket will contain uploaded files, temporary files, and job results.

Cancel

Save

Figure: AWS Storage page

The following settings apply to S3 access.

NOTE: The values that you should use for these settings should be provided by your S3 administrator. If they have already been specified, do not modify unless you have been provided instructions to do so.

Setting	Description
AWS Access Key	This key defines the account to use to connect to AWS.
AWS Secret Key	The secret (or password) associated with the key.
Default S3 Bucket	<div>This bucket is used for storage, unless another bucket is explicitly selected.<div>NOTE: Specify the top-level bucket name only. There should not be any backslashes in your entry.</div></div>
Extra S3 Buckets	You can specify a comma-separated string of additional S3 buckets that are available for storage. Do not put any quotes around the string. Whitespace between string values is ignored.

Transfer Asset Ownership

If necessary, you can transfer the ownership of one or more of your assets to another user.

Asset ownership can be transferred for the following types of assets:

- Flows

NOTE: You cannot transfer ownership of folders.

NOTE: Reference datasets cannot be separately transferred. They are transferred when the entire flow is transferred.

- Plans
- Imported datasets
- Connections

NOTE: When a connection is transferred to a new owner, the new owner may need to enter authentication credentials.

- Macros

NOTE: When a macro is transferred to a new owner, the previous owner loses access to it.

After the transfer, the original owner retains access to the assets as a collaborator.

NOTE: After a macro is transferred, the original owner no longer has collaborator access to the macro, since macros cannot be shared.

Notes:

- Administrators can transfer all assets owned by one user to another user, which is an important step before disabling or removing the original user from the system. For more information, see *User Details Page*.
- Asset ownership cannot be transferred between projects or workspaces.
- Transferring ownership of an asset does not change any underlying permissions that may be required to access the asset. For example, if you transfer an imported dataset to another user, that user cannot use the asset unless the user's permissions enable access. For more information, please contact your administrator.

Tip: Non-admin users can transfer only assets that they own. Admin users can transfer ownership of any Trifacta asset in the project or workspace.

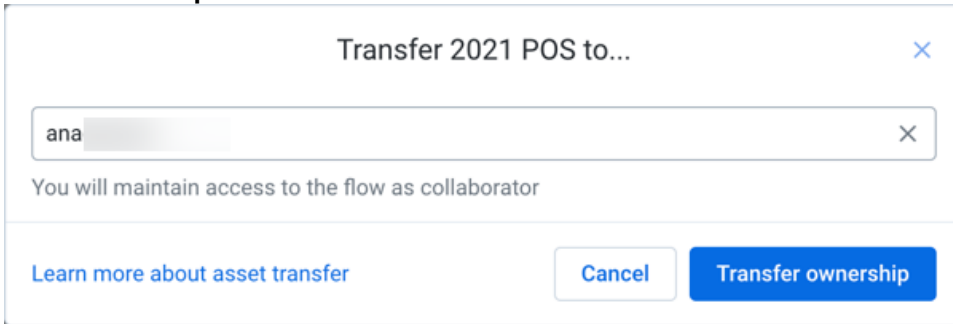
Transfer of Individual Assets

You can transfer the ownership of one or more individual assets.

Steps:

1. In the Designer Cloud application, locate the asset that you wish to transfer.
2. From the context menu for the asset, select **Transfer asset ownership**.

3. Enter the username of the individual to whom you wish to transfer ownership of the asset. Then, select **Transfer ownership**.



The dialog box is titled "Transfer 2021 POS to..." and has a close button (X) in the top right corner. Below the title is a text input field containing the username "ana" and a clear button (X) on the right. Below the input field is the text "You will maintain access to the flow as collaborator". At the bottom of the dialog, there are three buttons: a blue link "Learn more about asset transfer", a white "Cancel" button, and a blue "Transfer ownership" button.

Figure: Transfer asset ownership dialog

4. The asset is now owned by the designated user.

Bulk Transfer

Admin users can transfer all of the assets owned by one user to another user. For more information, see *User Details Page*.

Non-admin users can transfer all of their owned assets through the API. See <https://api.trifacta.com/ee/es.t/index.html#tag/Workspace/operation/transferUserAssetsInCurrentWorkspace>

Admin

Contents:

- *Test Product by Running a Job*
- *Integrate the Product*
 - *Designer Cloud Powered by Trifacta Enterprise Edition*
- *Review Environment Settings*
- *Invite Users*
- *Modify Roles*

After initial deployment, most admin work for the Designer Cloud powered by Trifacta® platform can be managed through the Designer Cloud application, including user provisioning, data access, and project or workspace configuration settings.

Tip: If you are a project owner or workspace admin logging into the Designer Cloud application for the first time, you can use these steps to set up the environment for use and invite other users to join.

Test Product by Running a Job

When you first log in to your project or workspace, you should be able to immediately run a job to ensure the product is working properly. The goal of these steps is to simply verify that you can run a job.

Tip: Any user invited to the project should be able to complete these steps, including uploading files from their local environment to begin wrangling immediately.

Steps:

1. In the left nav bar, click the **Flows** icon at the top.
2. In the Flows page, you should see an `Example Flows` folder.
 - a. If you do, click it to open it.
 - b. If you do not, you can:
 - i. Click the **Library** icon. Then, click **Import Data**.
 - ii. Upload a file. For more information, see *Import Data Page*.
 - iii. Then, continue the process of adding the file to a flow and running a job from there.
3. Select either of the flows.
4. The flow opens in Flow View. Click the **Plus** icon next to one of the recipe objects in the flow canvas. Select **Create Output to run**.
5. An output object is created. This output defines how job results are published. In this case, the default output is a CSV file in the default location. In the right panel, click **Run**.
6. In the Run Job page, you can review the options. For simplicity, accept the defaults. Click **Run**.
7. The job is queued for execution. The Job Details page permits you to track progress.
8. When the job completes, click the **Output destinations** tab to review your outputs.

If you have successfully completed the above steps, the product is working for end-to-end execution of importing, transforming, and outputting your data.

Integrate the Product

Depending on the edition of the product that you licensed, there may be specific steps required to integrate the Designer Cloud powered by Trifacta platform with your environment:

Tip: Please complete the steps listed below for your product edition by following the documentation link. You can then return to complete the remaining steps in this page.

Designer Cloud Powered by Trifacta Enterprise Edition

Key Tasks:

1. Installation of the product on an edge node of your cluster.
2. Configuration of the product to integrate with the clustered running environment.
3. Additional configuration settings may be required.

For more information:

- *Install*
- *Configure*

Review Environment Settings

After you have deployed the product, you should review the environment settings.

Steps:

1. Login to the Designer Cloud application as an administrator.
2. Select **User menu > Admin console**.
3. Select **Project Settings** or **Workspace Settings**.

These settings define features and behaviors in the project or workspace. Key categories and settings:

Category	Notes
API	<p>These settings define whether users are permitted to create and use API access tokens, which allow for access to the REST APIs.</p> <p>Tip: API access is required for developers who wish to build on the platform or users who wish to automate aspects of their data pipelines.</p>
Connectivity	<p>These features can enable access to datastores and conversion features, including the use of custom SQL to create imported datasets.</p> <p>Tip: If significant volumes of your data are hosted in relational sources, you should review these settings.</p>
Flows, recipes, and plans	<p>These settings enable features related to the development of features, recipes, and plans.</p> <p>Tip: By default, users are permitted to import, export, and share flows and plans, as well as create webhooks to deliver messages outside of the product. If these features need to be disabled, please review these settings.</p>
Job execution	<p>These features define aspects of how jobs are executed on Trifacta Photon, which is an in-memory running environment.</p> <p>Tip: Most of these settings are advanced tuning properties. Trifacta Photon may require enablement in your environment.</p>
Publishing	<p>These settings can be modified to define the formats that Designer Cloud powered by Trifacta platform is permitted to generate. Most output formats are enabled.</p>

Experimental features

These features are early access features that may be modified or even removed at any time.

Tip: To begin, you should avoid enabling Experimental features until you are familiar with the product.

Invite Users

You can now invite users to your project or workspace. See *Invite Users*.

Modify Roles

Each invited user is automatically assigned the default role. If needed, you can modify or add other roles to the user account.

NOTE: Without modification, the default role assigned to users permits sufficient access to import, transform, and export data. Access to admin functions and other advanced features may be restricted.

Tip: Roles are additive. Users are permitted the maximum privileges in all assigned roles.

Steps:

1. In the Users, find the user to modify. Click the **More (...)** menu and select **Edit**.
2. Select roles from the Roles drop-down.
3. Then, click **Edit user**.

As needed, you can modify the privileges of existing roles or define new roles.

Admin Tasks

This section covers administrative tasks for configuring storage and running environments, setting up the basic parameters and preferences, creating IAM roles, registering users, and assigning levels of privileges.

Operations Tasks

This section contains administrative tasks related to the operations of the Designer Cloud powered by Trifacta® platform , including job execution within your enterprise ecosystem.

Configure Running Environments

Contents:

- *Trifacta Photon*
 - *EMR*
 - *Snowflake*
 - *Other Running Environments*
-

This section provides overview information on how to configure the running environments accessible from your deployment of the Designer Cloud application .

A **running environment** is the set of services that are used to execute a job.

- A job can include tasks to do the following:
 - Ingest data
 - Transform data
 - Profile data
 - Sample data
 - Generate results
- A running environment can be hosted on the Trifacta node or across a cluster that is connected to the product.

Trifacta Photon

Hosted on the Trifacta node, Trifacta Photon is an in-memory running environment designed for high performance on small- to medium-sized jobs.



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Configuration:

Trifacta Photon may require enablement in your project or workspace:

- For more information, see *Workspace Settings Page*.

EMR

Amazon Elastic Map Reduce (EMR) is a managed-cluster data platform for processing large volumes of disparate sources of data. This scalable platform is used for running jobs and can handle data processing tasks of any size.

Configuration:

- The Designer Cloud application must be connected to an EMR cluster. For more information, see *Configure for EMR*.
- If you are accessing AWS resources using IAM roles, those roles must contain policies to run jobs on EMR. For more information, see *Required AWS Account Permissions*.

Snowflake



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Snowflake provides cloud-based data storage and analytics as a service. If all of your source datasets and outputs are in Snowflake locations and other conditions are met, then the entire execution of the transformations can occur in Snowflake. For more information, see <https://www.snowflake.com>.

For datasets and outputs that are hosted in Snowflake, you can configure the Designer Cloud application to perform the transformation steps of your job in Snowflake. In this manner, no data needs to be transferred to and from the data warehouse, and performance should be significantly better.

Tip: Jobs must be enabled for execution in Snowflake for each flow. For more information, see *Flow Optimization Settings Dialog*.

Limitations:

NOTE: Snowflake is not a running environment that you explicitly select or specify as part of a job. If all of the requirements are met, then the job is executed in Snowflake. For more information on limitations, see *Overview of Job Execution*.

Configuration:

For more information, see *Snowflake Running Environment*.

Other Running Environments

Depending on your deployment of the Designer Cloud powered by Trifacta platform, additional running environments may be available. For more information, see *Running Environment Options*.

Verify Operations

Contents:

- *Prepare Your Sample Dataset*
 - *Store Your Dataset*
 - *Import Dataset*
 - *Run Job*
 - *Review Results*
-

After you have applied a configuration change to the platform and restarted, you can use the following steps to verify that Designer Cloud Powered by Trifacta® Enterprise Edition is working correctly.

If your configuration change was applied to `trifacta-conf.json`, you should restart the platform before continuing. See *Start and Stop the Platform*.

Prepare Your Sample Dataset

To complete this test, you should locate or create a simple dataset. Your dataset should be created in the format that you wish to test.

Tip: The simplest way to test is to create a two-column CSV file with at least 25 non-empty rows of data. This data can be uploaded through the application.

Characteristics:

- Two or more columns.
- If there are specific data types that you would like to test, please be sure to include them in the dataset.
- A minimum of 25 rows is required for best results of type inference.
- Ideally, your dataset is a single file or sheet.

Store Your Dataset

If you are testing an integration, you should store your dataset in the datastore with which the product is integrated.

Tip: Uploading datasets is always available as a means of importing datasets.

- You may need to create a connection between the platform and the datastore.
- Read and write permissions must be enabled for the connecting user to the datastore.
- For more information, see *Connections Page*.

Import Dataset

Steps:

1. Login to the application.
2. In the application menu bar, click **Library**.
3. Click **Import Data**. See *Import Data Page*.
 - a. Select the connection where the dataset is stored. For datasets stored on your local desktop, click **Upload**.

- b. Select the dataset.
- c. Click **Continue**.

Run Job

Run a job to generate results.

Steps:

1. The initial sample of the dataset is opened in the Transformer page, where you can edit your recipe to transform the dataset.
 - a. In the Transformer page, some steps are automatically added to the recipe for you. So, you can run the job immediately.
 - b. You can add additional steps if desired. See *Transformer Page*.
2. Click **Run**.
 - a. If options are presented, select the defaults.
 - b. To generate results in other formats or output locations, click **Add Publishing Destination**. Configure the output formats and locations.
 - c. To test dataset profiling, click the Profile Results checkbox. Note that profiling runs as a separate job and may take considerably longer.
 - d. See *Run Job Page*.

Review Results

1. When the job completes, you should see a success message under the Jobs tab in the Flow View page.
 - a. **Troubleshooting:** Either the Transform job or the Profiling job may break. To localize the problem, try re-running a job by deselecting the broken job type or running the job on a different running environment (if available). You can also download the log files to try to identify the problem. See *Job Details Page*.
2. Click **View Results** from the context menu for the job listing. In the Job Details page, you can see a visual profile of the generated results. See *Job Details Page*.
3. In the Output Destinations tab, click a link to download the results to your local desktop.
4. Load these results into a local application to verify that the content looks ok.

Checkpoint: You have verified importing from the selected datastore and transforming a dataset. If your job was successfully executed, you have verified that the product is connected to the job running environment and can write results to the defined output location. Optionally, you may have tested profiling of job results. If all of the above tasks completed, the product is operational end-to-end.

Access Management Tasks

This section contains admin tasks to configure and maintain access to your enterprise datastores from the Designer Cloud powered by Trifacta® platform .

Enable Access to S3 and AWS Resources

Contents:

- *AWS Overview*
- *Technical Setup*
 - *Create policy to grant access to S3 bucket*
 - *Update policy to accommodate SSE-KMS if necessary*
 - *Add policy for Redshift access*
 - *Whitelist the IP address range of the Trifacta Service, if necessary*

If you plan to use S3 as the default storage environment, the following sections outline the AWS configuration prerequisites and requirements.

Tip: This section should be shared with your S3 administrator, who can provide the required information.

AWS Overview

Below are the AWS objects that are required for S3 setup.

AWS object	Required?	Description
AWS account	Y	To create these objects are part of the setup process, you must have an AWS account. For more information, see https://aws.amazon.com/ .
Valid email address	Y	To validate your registration for a new workspace, you must have a valid email address to which the product can deliver the registration email.
Choice: cross-account role access or key-secret access	Y	<p>To integrate with your existing S3 resources, you must choose a method of authentication. Choices:</p> <ul style="list-style-type: none">• cross-account role: This method uses IAM roles to define the permissions used by the product for S3 access. <div>Tip: This method is recommended.</div> <ul style="list-style-type: none">• key-secret access: This method uses an IAM access keys to provide S3 access.
IAM policy	Y	<p>An IAM (Identity and Access Management) policy is an AWS resource used to define the low-level permissions for access to a specific resource. You can use an IAM policy for the product to use for either access method.</p> <p>For more information, see "Create policy to grant access to S3 bucket" below.</p>
cross-account role access: IAM role	Y	An IAM role contains one or more IAM policies that can be used to define the set of available AWS services and the level of access to them for a user. In this case, the user is the Designer Cloud application .
key-secret access: AWS key-secret	Y	An older AWS access method, the key-secret combination is essentially a username and password combination to one or more S3 buckets.
S3 bucket	Y	S3 (Simplified Storage Service) is a cloud-based file storage system hosted in AWS. An S3 bucket contains your data files and their organizing folders.
S3 bucket: encryption	N	<p>For better security, your S3 bucket may be encrypted, which means that the data is stored inside of S3 in a way that is not human-readable.</p> <div>NOTE: The product can optionally integrate with encrypted S3 buckets. The following S3 encryption methods are supported: sse-s3 and sse-kms.</div>

		<p>NOTE: If your bucket is encrypted with ss3-kms, additional configuration is required. See "Update policy to accommodate SSE-KMS if necessary" below.</p> <p>For more information on your bucket's encryption, please contact your S3 administrator.</p>
S3 bucket: storage location	N	<p>If needed, you can change the location where results are stored in S3.</p> <p>NOTE: The product must have write permission to this location. If you are changing the location from the default, please verify with your S3 administrator that the preferred location is enabled for writing through your access method.</p>
IAM role: Account ID	N	<p>The account ID identifies in the trust policy that Trifacta AWS account can use your IAM role.</p> <p>Tip: This identifier is provided to you during registration and setup.</p>
IAM role: External ID	N	<p>The external ID identifies in the trust policy that Designer Cloud application can use your IAM role only on your behalf.</p> <p>Tip: This identifier is provided to you during registration and setup.</p>

Technical Setup

The following sections should be provided to your AWS administrator for setting up access to these resources, if required.

Create policy to grant access to S3 bucket

To use your own S3 bucket(s) with Designer Cloud application, create a policy and assign it to either the user or IAM Role selected to grant access to AWS resources. In this section, you create the policy. Later, it will be applied.

- For more information on creating policies, see <https://console.aws.amazon.com/iam/home#/policies>.

Below is an example policy template. You should use this template to create the policy.

NOTE: You should not simply use one of the predefined AWS policies or an existing policy you have as it will likely give access to more resources than required.

Template Notes:

- One of the statements grants access to the public demo asset buckets.
- Replace `<my_default_S3_bucket>` with the name of your default S3 bucket.
- To grant access to multiple buckets within your account, you can extend the resources list to accommodate the additional buckets.

Policy Template

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
```

```

        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:ListBucket",
            "s3:DeleteObject",
            "s3:GetBucketLocation"
        ],
        "Resource": [
            "arn:aws:s3:::my_default_S3_bucket",
            "arn:aws:s3:::<my_default_S3_bucket>/*"
        ]
    },
    {
        "Sid": "VisualEditor1",
        "Effect": "Allow",
        "Action": [
            "s3:GetObject",
            "s3:ListBucket"
        ],
        "Resource": [
            "arn:aws:s3:::aws-saas-samples-prod",
            "arn:aws:s3:::aws-saas-samples-prod/*",
            "arn:aws:s3:::aws-saas-datasets",
            "arn:aws:s3:::aws-saas-datasets/*",
            "arn:aws:s3:::3fac-data-public",
            "arn:aws:s3:::3fac-data-public/*",
            "arn:aws:s3:::trifacta-public-datasets",
            "arn:aws:s3:::trifacta-public-datasets/*"
        ]
    }
]
}

```

Update policy to accommodate SSE-KMS if necessary

If any accessible bucket is encrypted with SSE-KMS, another policy must be deployed. See <https://docs.aws.amazon.com/kms/latest/developerguide/iam-policies.html>.

Add policy for Redshift access

If you are connecting to Redshift databases through your workspace, you can enable access by creating a `GetClusterCredentials` policy. This policy is additive to the the S3 access policies. All of these policies can be captured in a single IAM role.

Example:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GetClusterCredsStatement",
      "Effect": "Allow",
      "Action": [
        "redshift:GetClusterCredentials"
      ],
      "Resource": [
        "arn:aws:redshift:us-west-2:123456789012:dbuser:examplecluster/${redshift:DbUser}",
        "arn:aws:redshift:us-west-2:123456789012:dbname:examplecluster/testdb",
        "arn:aws:redshift:us-west-2:123456789012:dbgroup:examplecluster/common_group"
      ],
      "Condition": {
        "StringEquals": {
          "aws:userid": "AIDIODR4TAW7CSEXAMPLE:${redshift:DbUser}@yourdomain.com"
        }
      }
    }
  ]
}

```

```
}  
]  
}
```

For more information on these permissions, see *Required AWS Account Permissions*.

Whitelist the IP address range of the Trifacta Service, if necessary

If you are enabling any relational source, including Redshift, you must whitelist the IP address range of the Trifacta service in the relevant security groups.

NOTE: The database to which you are connecting must be available from the Trifacta service over the public Internet.

The IP address range of the Trifacta service is:

```
35.245.35.240/28
```

For Redshift:

For Redshift, there are two ways to whitelist the IP range depending on if you are using EC2-VPC or EC2-Classic (not common).

- **EC2-VPC (Security group):** Add the IP address range to the inbound rule for the security group associated with the cluster. For more information, see <https://docs.aws.amazon.com/redshift/latest/gsg/rs-gsg-authorize-cluster-access.html#rs-gsg-how-to-authorize-access-vpc-security-group>
- **EC2-Classic:** Add the IP address range to the inbound rule for the security group associated with the EC2 instance. For more information, see <https://docs.aws.amazon.com/redshift/latest/gsg/rs-gsg-authorize-cluster-access.html#rs-gsg-how-to-authorize-access-cluster-security-group>

For details on this process with RDS in general, see <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.RDSSecurityGroups.html>

For more information, please contact *Alteryx Support*.

Insert Trust Relationship in AWS IAM Role

If you are using per-user authentication through an AWS IAM role, you must insert a trust relationship into the role so that the Designer Cloud powered by Trifacta® platform can leverage it.



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Prerequisites:

NOTE: These steps should be performed by an AWS administrator.

Please acquire the following information:

- **IAM role:** The AWS IAM role that the Designer Cloud powered by Trifacta platform should use.
- **EC2 instance role:** If the EC2 instance role is to be used to assume the AWS role, then please acquire the following:
 - AWS account ID
 - EC2 instance role
 - Details on the above are listed below.

For more information on the AWS Principal options described below, please review https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_elements_principal.html.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameter and retrieve its value (`true` or `false`):

```
"aws.ec2InstanceRoleForAssumeRole"
```

3. Login to the AWS console.
4. Open the IAM role for use with the Designer Cloud powered by Trifacta platform .
5. If `aws.ec2InstanceRoleForAssumeRole=true`, then the EC2 instance role is used for assuming the provided AWS role. Paste the following into the IAM role for the trust relationship:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::<awsAccountId>:role/<ec2InstanceRole>"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Property	Description
<awsAccountId>	AWS account identifier for which the EC2 instance role is assumed

<ec2InstanceRole>	EC2 instance role to use
-------------------	--------------------------

6. If `aws.ec2InstanceRoleForAssumeRole=false`, then the AWS user associated with the provided AWS key and secret is assumed. Paste the following into the IAM role for the trust relationship:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::862753480162:user/sample-user"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

7. Save the IAM role definition.

User Admin Tasks

This section contains admin tasks on provisioning and maintaining users, their roles, and their permissions within the Designer Cloud powered by Trifacta® platform .

Create User Account

Contents:

- *Creating your own user account*
 - *Creating users when self-registration is disabled*
 - *Troubleshooting*
 - *Account Not Configured login error*
-

By default, users can create their own accounts. As needed, self-registration can be disabled, so that all users must be created by an administrator. See *Configure User Self-Registration*.

Creating your own user account

Steps:

1. Users may self-register at the following address:
`http://<host_name>:<port_number>`
where:
`<host_name>` is the host of the Designer Cloud® application .
`<port_number>` is the port number to use. Default is 3005.
2. Click the Register link.
3. Enter your credentials in the spaces provided. A valid email address is required.
4. As soon as the account is created, you may login at the first address. See *Login*.

Creating users when self-registration is disabled

When self-registration is disabled, an administrator must manually create the accounts for users. Administrators can create accounts at the following address:

`http://<host_name>:<port_number>/register`

NOTE: If SSO or secure impersonation is enabled in your environment, administrators must apply a principal value to each newly created user. See *Users Page*.

When a new account is created, an email is sent to the address for the created user.

Troubleshooting

Account Not Configured login error

If you have created a user account, you may see the following error message when you try to login:



Account Not Configured

Your Trifacta user account has not been completely configured.

[Configure storage settings](#) or contact your Trifacta Administrator.

[Return to Sign In](#)

Figure: Account Not Configured

In this case, the account may require additional configuration. In SSO or Kerberos environments, an administrator may need to provision a SSO or Hadoop principal value to be applied to the user account. See *Admin Settings Page*.

Configure Password Criteria

By default, the Designer Cloud® application enforces very few requirements on password length, capitalization, or special characters. Users who are setting or resetting their passwords are permitted to create a password of one character in length with no additional requirements.

NOTE: When passwords are set or reset, the platform does perform an assessment of the quality of the password and reports it to the user before saving. For more information, see *User Profile Page*.

Before you permit users to create accounts, you should review the password requirements for your enterprise and, where needed, apply them to the Designer Cloud application .

Enable

To enable enforcement of password criteria, please enable the following parameter.

Steps:

You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.

Locate the following parameter and set it to `true`:

```
"feature.enablePasswordCriteria": true,
```

When enabled, submitted changes to user passwords are evaluated based on the configuration settings defined below.

Configure

The following parameters govern the password criteria enforced by the Designer Cloud application when the feature is enabled.

Parameter	Description	Default
webapp.passwordCriteria.length.min	Minimum length of a password to Designer Cloud application	0
webapp.passwordCriteria.length.max	Maximum length of a password to Designer Cloud application	100
webapp.passwordCriteria.description	Text describing the criteria that a password must meet. Specify this value last.	
webapp.passwordCriteria.contains.uppercase	Defines whether the password must contain uppercase characters	undefined
webapp.passwordCriteria.contains.symbols	Defines whether the password must contain symbols	undefined
webapp.passwordCriteria.contains.spaces	Defines whether the password must contain space characters	undefined
webapp.passwordCriteria.contains.lowercase	Defines whether the password must contain lowercase characters	undefined
webapp.passwordCriteria.contains.letters	Defines whether the password must contain letters (a-z)	undefined
webapp.passwordCriteria.contains.digits	Defines whether the password must contain digits (0-9)	undefined

Criteria settings:

Some of the criteria settings support the following options:

Setting	Description
enforce	Each password must pass this requirement.
forbid	Passwords cannot have this requirement.
undefined	(default) This requirement is disabled. Users may choose to include or not include this requirement in their passwords.

Create Admin Account

Contents:

- *Default admin account*
- *Create admin accounts*
- *Create admin account outside the UI*
 - *Without SSO*
 - *With SSO*

You can create additional administrator accounts to the Designer Cloud powered by Trifacta® platform using one of the following methods.

Default admin account

When the Designer Cloud powered by Trifacta platform is installed, a default admin account is created for you. For licensing purposes, this account is counted as a valid user.

The password for the default admin account should be changed as soon as you have access to the application. See *Change Admin Password*.

NOTE: Do not delete the default admin account. To ensure that there is always one admin account that is accessible, this account is automatically recreated if you delete it.

NOTE: Since this account cannot be mapped to a valid email address within a customer domain, it cannot be used in an SSO environment.

Create admin accounts

Steps:

1. Login using another admin account.
2. Create the account normally. See *Create User Account*.
3. Select **User menu > Admin console > Users**.
4. For the newly created user, select **Edit** from the user's context menu.
5. Admin roles:
 - a. To enable administration of workspace users, roles and other settings, select **Workspace admin** from the Roles drop-down.
 - b. To enable administration of platform settings, click the **Platform admin** checkbox.
6. Save changes.
7. Login to the account and verify that the Admin console pages are available.

Create admin account outside the UI

If you do not have access to an admin account through the application, you can create admin accounts for users from the Trifacta node using the `webapp/bin/ensure-user` command.

Without SSO

If Single Sign-On (SSO) is not enabled, use the following command:

```
<install_dir>/webapp/bin/ensure-user --admin "<FirstName LastName>" <e-mail> <password>
```

With SSO

If the environment uses SSO, the following command can create the admin user based on an Active Directory login:

```
<install_dir>/webapp/bin/ensure-user --admin "<FirstName LastName>" <e-mail> <password> <AD_LOGIN>
```

where:

<AD_LOGIN> is the active directory login for the user.

Manage Users under SSO

Contents:

- *Enable SSO*
- *Configure Auto-Registration*
 - *User Management with Auto-Registration*
 - *Disable Auto-Registration*
 - *Provision new users under SSO without auto-registration*
 - *User access for reverse proxy method*

This section covers additional requirements for managing users of the Designer Cloud powered by Trifacta® platform in SSO environments.

Enable SSO

The Designer Cloud powered by Trifacta platform requires additional configuration to integrate with your SSO provider. Available methods:

Method	Description
SAML IDP	Integrate the platform with enterprise SAML identity provider. See <i>Configure SSO for SAML</i> .
Native LDAP-AD	Using native functionality in the platform, it can integrate with enterprise LDAP/AD. For more information, see <i>Configure SSO for AD-LDAP</i> .
LDAP-AD via reverse proxy	<div>A reverse proxy server outside of the platform can be set up for integration with enterprise LDAP/AD.</div> <div>NOTE: This method is likely to be deprecated in a future release.</div> <div>For more information, see <i>Configure SSO for AD-LDAP</i>.</div>

Configure Auto-Registration

Tip: By default, user auto-registration is enabled. It is recommended.

How users are managed depends on whether auto-registration is enabled:

- If auto-registration is enabled, after users provide their credentials, the account is automatically created for them.
- If auto-registration is disabled, a Trifacta administrator must still provision a user account before it is available. See below.

User Management with Auto-Registration

After SSO with auto-registration has been enabled, you can still manage users through the Designer Cloud application , with the following provisions:

- The Designer Cloud powered by Trifacta platform does not recheck for attribute values on each login. If attribute values change with your identity provider, they must be updated in the configuration.
 - For more information, see *Configure SSO for AD-LDAP*
 - For more information, see *Configure SSO for SAML*.

- If the user has been removed from AD, the user cannot sign in to the platform.
- If you need to remove a user from the platform, you should just disable the user through the Designer Cloud application .
 - If the user is deleted, then if the user returns to the platform in the future, a new account is created for the user.

For more information, See *Users Page*.

Disable Auto-Registration

To disable auto-provisioning in the platform, please verify the following property:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Set the following property:

```
"webapp.sso.enableAutoRegistration" : false,
```

3. Save your changes and restart the platform.
4. New users of the Designer Cloud powered by Trifacta platform must be provisioned by a Trifacta administrator. See below.

Provision new users under SSO without auto-registration

If SSO auto-registration is disabled, admin users can provision new users of the platform through the following URL:

```
https://<hostname>:<sso_port_number>/register
```

where:

- `<hostname>` is the host of the Designer Cloud powered by Trifacta platform
- `<sso_port_number>` is the port number.

The user's password is unnecessary in an SSO environment. You must provide the SSO principal value, which is typically the Active Directory login for the user.

- If you are connected to a Hadoop cluster, you must provision the Hadoop principal value.
- See *Create User Account*.

User access for reverse proxy method

Users access the application through the Trifacta node using the standard hostname and the port that you specified:

NOTE: All users must be use this URL to access the Designer Cloud application . If they use the non-SSO URL, they may receive an `Unprovisioned User` error.

```
https://<hostname>:<sso_port_number>
```


Invite Users

Contents:

- *Invite User*
 - *Edit User*
 - *Edit roles*
 - *Change workspace admin*
 - *Assign roles*
 - *Disable User*
 - *Remove User*
-

Administrators can manage the users who are permitted to use the Designer Cloud powered by Trifacta® platform .

All of these functions are available through the Admin console. For more information, see *Admin Console*.

Invite User

To permit a user to access the Designer Cloud application , an administrator must complete the following steps.

NOTE: When a user accepts your invitation, the additional user counts toward the maximum number of permitted users.

NOTE: If you are re-inviting a user who has been removed, you must wait 14 days to invite the user back to the same project or workspace and retain the user's data. If restoring the user's assets is not important, please contact *Alteryx Support* for immediate re-instatement.

Steps:

1. Login to the Designer Cloud application as an administrator.
2. From the left navigation bar, select **User menu > Admin console > Users**.
3. In the Users page, click **Invite users**.
4. In the Invite users dialog, enter a comma-separated list of email addresses to which to send invites.
 - a. These addresses become the user identifier for logging into the Designer Cloud application .
 - b. Avoid sending invites to email aliases.
 - c. Example:

```
joe.smith@example.com, mary.jones@example.com
```

5. To invite the list of users, click **Invite users**.
6. An email is sent to each valid user email address that you listed. The receiving user must click the link in the email to accept the invitation.

The user is invited via email and created in the Designer Cloud application . You can modify the user account as needed before the user chooses to log in. See below.

For more information, see *Users Page*.

Edit User

Edit roles

Steps:

1. In the Users page, locate the user to review.
2. On the right side of the row for the user, click the Actions menu.
3. Select **Edit user**.
4. In the dialog, you can add and remove roles for the user account.
5. When finished, click **Edit User**.

Change workspace admin

By default, a new user account is assigned a non-admin role. If needed, you can assign the user to be a workspace admin.

Use the following steps to change a user's workspace role between non-admin and admin.

Steps:

1. In the Users page, locate the user whom you are promoting to admin.
2. On the right side of the row for the user, click the Actions menu.
3. Select **Change admin role**.
4. In the Change admin role dialog, select the workspace role:
 - a. **Member** - standard user account, which is not permitted access to the Admin console and its functions.
 - b. **Admin** - administrator account, which can access all available features of the workspace.

You should avoid assigning the admin role to a large number of users.

5. Click **Save**.
6. The user's workspace role is immediately updated.

Assign roles

When the account is created, it is automatically assigned the `Default` role. You should review the permissions associated with this role and to determine if the user needs to be assigned a different one. For more information, see *Roles Page*.

Disable User

NOTE: Disabled users still count toward workspace limits on number of users.

If needed, a user's account can be disabled from accessing the Designer Cloud powered by Trifacta platform . When a user account is disabled:

- The user can no longer log in to the Designer Cloud application or use any available API endpoints.
- The user's assets in Designer Cloud application are retained. They can be accessed by other users who have been granted permission.

To disable a user, please complete the following steps:

Steps:

1. In the Users page, locate the user to disable.
2. On the right side of the row for the user, click the Actions menu.
 - a. To disable **Disable**. Click **Disable** to confirm.
 - b. To reactivate a disable member, click **Enable**.
3. Effective immediately, the user cannot log in to the application.

Remove User

To remove a user completely, please complete the following steps.

When a user is removed from the Designer Cloud powered by Trifacta platform , any assets that are owned by the user must be reassigned to other users, or they are lost and no longer accessible even by an administrator.

Steps:

1. In the Users page, locate the user to remove.
2. On the right side of the row for the user, click the Actions menu.
3. Select **Remove**.
4. If the user owns assets, you can choose to assign them to another user. If you do not assign them, these assets are lost.
5. Confirm that you wish to remove the user.

If you must recover a removed user or that user's assets, please contact *Alteryx Support* within 14 days of the deletion.

Create Role

Contents:

- *Create Role*
 - *Example - Read-only access role*
 - *Example - Flows-only access role*
 - *Example - Empty role*
 - *Assign Role*
 - *Modify Role*
 - *Example - Modify default role*
 - *Unassign Role*
 - *Delete Role*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Administrators can create and assign roles to users to govern access to user-created objects in the Designer Cloud powered by Trifacta® platform .

- A **role** is a set of privileges that can be assigned to users.
- A **privilege** governs access level to a type of object.
- By default, all users are assigned the `default` role, which allows users to use the user-created object types.
- For more information, see *Privileges and Roles Reference*.

As needed, you can create user roles to define different access levels for different object types.

NOTE: You must be an administrator to create new roles.

NOTE: Roles are additive. If you assign multiple roles to a user account, the user receives the highest level of access for each privilege among the assigned roles.

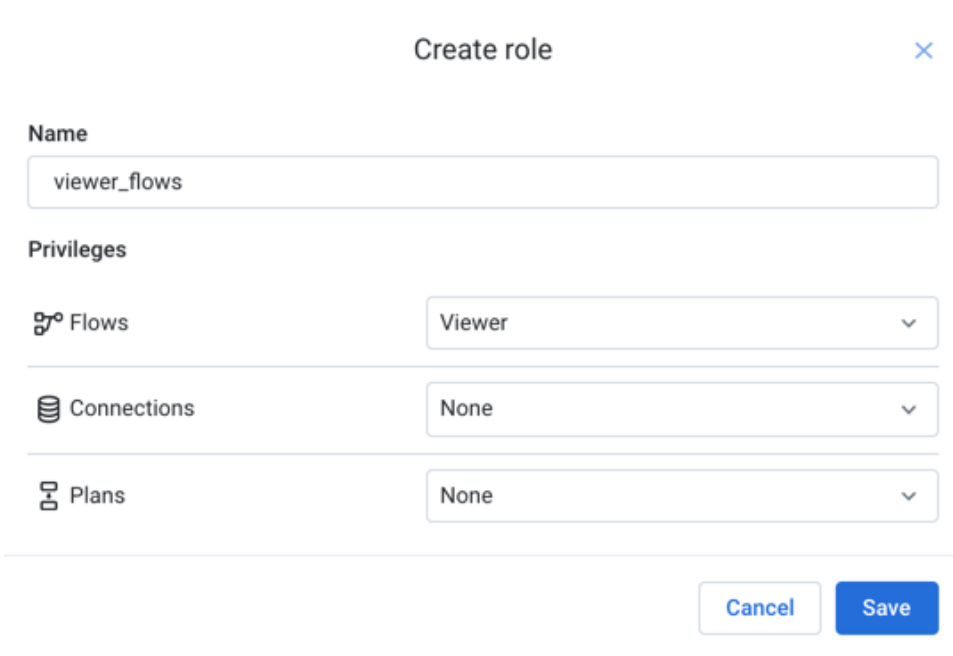
NOTE: When a role is assigned, unassigned, or modified, the changes to privileges are immediately applied to the associated user accounts. A new login is not required.

Create Role

To create a new role, please complete the following steps.

Steps:

1. In the left nav bar, select **User menu > Admin console > Roles**.
2. In the Roles page, review the list of available roles. For more information, see *Roles Page*.
3. To create a new role, click **Create Role**.



The image shows a 'Create role' dialog box. At the top, it says 'Create role' with a close button (X) on the right. Below this is a 'Name' field containing the text 'viewer_flows'. Underneath is a 'Privileges' section with three rows. The first row is 'Flows' with a 'Viewer' dropdown menu. The second row is 'Connections' with a 'None' dropdown menu. The third row is 'Plans' with a 'None' dropdown menu. At the bottom right of the dialog are two buttons: 'Cancel' and 'Save'.

Figure: Create Role dialog

4. In the Create Role dialog, specify the following:
 - a. **Name:** Enter a name for your role. This value must be unique among available roles.
 - b. **Privileges:**
 - i. For each of the available object types, specify the access level for the role.
 - ii. For more information on these privileges, see *Privileges and Roles Reference*.
 - c. To create the role, click **Save**.
5. The role is now available and can be assigned to users. See below.

For more information, see *Create Role Dialog*.

Example - Read-only access role

Suppose you wish to limit a set of users to read-only access to role-based objects.

Steps:

1. In the Roles page, click **Create role**.
2. In the Create Role dialog, enter the following:
 - a. Name: `read-only`
 - b. Privileges: For each available privilege, select `viewer`.

NOTE: Some privileges may not have a `viewer` access level. For these privileges, you should select `none`. However, users with such a role cannot access the pages where these objects are listed.

3. Click **Save**.
4. The role is now available and can be assigned to users. See below.

Example - Flows-only access role

Suppose you wish to limit a set of users to only be able to work with flows. These users should be able to view, share, edit, schedule, run jobs, and delete flows.

Steps:

1. In the Roles page, click **Create role**.
2. In the Create Role dialog, enter the following:
 - a. Name: `flows-only`
 - b. Privileges:
 - c. For the flows privilege, select `author`.
 - d. For every other privilege, select `none`.
3. Click **Save**.
4. The role is now available and can be assigned to users. See below.

Example - Empty role

In some circumstances, you may wish to assign an empty role to a user. For example, you may wish to limit some administrators to only be able to configure aspects of the platform without providing access to any user-created objects.

Steps:

1. In the Roles page, click **Create role**.
2. In the Create Role dialog, enter the following:
 - a. Name: `empty`
 - b. Privileges:
 - c. For every privilege, select `none`.
3. Click **Save**.
4. The role is now available and can be assigned to users. See below.

Assign Role

After a role has been created, you can assign it to users.

NOTE: Assigning a role adds the role to the user's account. It does not replace any role that is already present in the account.

NOTE: When you assign or unassign a role, the privileges are immediately applied to the assigned user's account. The user does not need to re-login to see the changes.

Steps:

1. In the Roles page, locate the role to assign.
2. On the right side of the screen, click the context menu for the role. Select **Assign role....**
3. In the Assign role dialog, enter a list of email addresses for users to whom you wish to assign the role.
4. Click **Assign**.
5. The role and its associated privileges are applied immediately to the user account(s).

Modify Role

After a role has been created, you can modify it as needed.

NOTE: When the privileges of a role are modified, the changes are applied immediately to all users who are currently assigned the role. Before making modifications, you should review the users who could be affected. See *Role Details Page*.

Steps:

1. In the Roles page, locate the role to modify.
2. In the context menu on the right side of the page, select **Edit**.
3. Review the privileges assigned to the role, and make any changes as necessary.
4. Click **Save**.
5. All users who currently have the role in their account immediately receive the changed privileges.

Example - Modify default role

Tip: If you are changing the privileges of a role, you might want to create a role that contains only the replaced privileges. For example, if you are changing access to flows in Role A from `author` to `editor`, you might create a new role first, which contains only the `author` privilege for flows. If a user needs to be able to create new flows, you can then assign the new role accordingly.

Suppose you wish to reduce privileges for the `default` role, which is assigned to all users. At the same time, some user should be assigned author-level access to the available objects. Here is the following general flow for managing this modification.

NOTE: You cannot modify the name of the `default` role.

Steps:

1. Before you begin, you might wish to inform users that you are making these changes. In some cases, users may lose access to objects that they have created.
2. Create new roles for author access to each object type. For more information, see *Roles Page*.
 - a. For example, you can create the `Flow Author` role, which has `author` privilege for flows and no other privilege. Optionally, for the other privileges, you could provide `viewer` access, which enables read-only access.
 - b. Repeat the above for each type of object for which there is a privilege.
 - c. At this point, the new roles have been created.
3. Assign these roles to users as needed. For example, for the `Flow Author` role, you can assign it to each user that must create flows.

Tip: Since roles are additive, you have not removed any privileges yet.

4. Now, you can modify the `default` role.
 - a. In this case, you should decide what is the baseline set of privileges that each new user should have. Set the privileges to the lowest level of common access.

Unassign Role

Use the following steps to remove a role from a user account.

NOTE: Removing a role from a user account may remove access to objects that the user has created. If the user is the owner of these objects, some access may be removed permanently, even if the object is shared. For more information, see *Overview of Sharing*.

Steps:

1. In the Roles page, locate the role that you wish to remove from one or more user accounts.
2. Select the role.
3. In the Role Details page, click the Users tab.
4. Locate the user to un-assign the role. In the context menu for the user, select **Unassign from role**.

5. The user no longer has the role in the account.

For more information, see *Role Details Page*.

Delete Role

You are permitted to delete roles that are still assigned to users. Deleting a role removes the role from all user accounts and cannot be undone. Before you delete a role, you should review the list of affected users and the objects to which they have access.

Steps:

1. In the Roles page, locate the role to delete. In the context menu, select **Delete**.
2. Confirm the deletion.
3. The role is deleted. All users who had the role can no longer access the privileges assigned in the role.

See *Roles Page*.

Application Management Tasks

This section describes how to manage specific aspects of the Designer Cloud application .

Manage Downloading

Contents:

- *Job Results*
 - *Samples*
 - *Flows and Plans*
 - *Imported Datasets*
 - *Dataset previews*
-

For security reasons, you may need to apply controls to the ability of users to download data from the Designer Cloud powered by Trifacta® platform . You can choose the types of downloading you limit using the settings in the sections listed below.

Job Results

The Designer Cloud application allows users to download their job results up to a pre-defined limit. If you set to this limit 0, job results cannot be downloaded at all from the application. When downloading job results is disabled, all job results must be downloaded outside of the Designer Cloud application .

Tip: This parameter can also be used to increase the maximum size of permitted downloads from the Designer Cloud application .

NOTE: In general, you should avoid downloading files that are larger than 1 GB in size from the Designer Cloud application . Other limits, such as timeout settings, may be applied, which can cause download failures. This setting is also applied for other hard limits within the Designer Cloud application , so please modify with caution. For large files, please try to download through your storage layer.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameter:

```
"webapp.maxQueryResultsSize": 1073741824,
```

3. Set the value to 0.
4. Save your changes and restart the platform.
5. Verify that job results cannot be downloaded. See *Job Details Page*.

Samples

In the Transformer page, users interact with samples of data from the entire dataset. New samples can be created as needed.

By default, samples can be downloaded to the local desktop. To disable sample downloading, please complete the following steps.

Steps:

1. You apply this change through the *Workspace Settings Page*. For more information, see *Platform Configuration Methods*.

2. Locate the **Sample downloads** parameter, and set it to `disabled`.
3. To verify:
 - a. Logout of the Designer Cloud application . Login again.
 - b. Check that the Download Sample as CSV context menu option is not available in the Recipe panel in the Transformer Page.

NOTE: Recipes can still be downloaded.

See *Recipe Panel*.

Flows and Plans

For backup and migration purposes, users can be permitted to download their flow and plan definitions.

NOTE: A flow definition does not contain any data from the referenced datasets.

If needed, you can disable the ability to export flows and plans from the Designer Cloud powered by Trifacta platform

Steps:

1. You apply this change through the *Workspace Settings Page*. For more information, see *Platform Configuration Methods*.
2. Locate the **Export** parameter, and set it to `disabled`.
3. To verify:
 - a. Logout of the Designer Cloud application . Login again.
 - b. Check that the context menu option for Export Flow is not available in Flow View. See *Flow View Page*.

Imported Datasets

The platform does not support downloading imported datasets.

Dataset previews

When you select an imported dataset, you can see a preview of the data. While no data is stored permanently on the local desktop, you may prefer to disable previews.

NOTE: When previews are disabled, selection of imported datasets requires that users know the contents based on filename, location, and size information.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Locate the following parameter:

```
"webapp.client.previewLoadLimit": 128000,
```

3. Set the value to 0.
4. Save your changes and restart the platform.
5. Verify that when you select a file or table to import, no preview of it is displayed. For more information, see *Import Data Page*.

Manage Schedules

Contents:

- *Enable or Disable Schedule*
- *Delete Schedule*
- *Create Schedule*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the Schedules page, administrators of Designer Cloud Powered by Trifacta® Enterprise Edition can manage all of the schedules in the deployment.

NOTE: The Schedules page is available to project owners and workspace administrators only.

Enable or Disable Schedule

Steps:

1. Login as an administrator.
2. In the left nav bar, click the Schedules icon.
3. In the Schedules page, locate the schedule to change.
4. For the schedule's entry, open the context menu on the right side of the page.
5. Select **Enable Schedule** or **Disable Schedule**.

Delete Schedule

Deleting a schedule cannot be undone.

Steps:

1. Login as an administrator.
2. In the left nav bar, click the Schedules icon.
3. In the Schedules page, locate the schedule to delete.
4. For the schedule's entry, open the context menu on the right side of the page.
5. Select **Enable Schedule** or **Disable Schedule**.

Create Schedule

A schedule is composed of:

- A schedule frequency
- A set of one or more scheduled outputs

These objects are created from within a flow in Flow View. See *Add Schedule Dialog*.

Manage Environment Parameters

Contents:

- *Create Environment Parameter*
 - *Use Environment Parameter*
 - *Limitations*
 - *Edit Environment Parameter Value*
 - *Delete Environment Parameter*
 - *Export Environment Parameters*
 - *Import Environment Parameters*
-

You can define parameters that are applicable across the entire project or workspace environment.

An **environment parameter** is a variable of String type defined by an administrator that any user of the environment can reference in their flows and flow-related objects.

NOTE: You must be a project owner or workspace administrator to manage environment parameters.

For more information on parameters, see *Overview of Parameterization*.

Create Environment Parameter

Steps:

1. A project owner or workspace administrator can select **User menu > Admin console > Environment parameters**.
2. In the Environment Parameters page, click **Create**.
3. Specify the parameter:
 - a. **Name:** Enter the display name for the parameter.

NOTE: All environment parameter names are automatically prepended with `env..`

- b. **Default value:** Enter the default value.

NOTE: The default value is stored as a String value.

4. To create another environment parameter, click **Add another**.
5. To save your changes, click **Save**.

Use Environment Parameter

Environment parameters can be referenced in the following locations:

Tip: When specifying a variable, enter `$env` to see the list of available environment parameters.

- Parameterized datasets

- Datasets created with SQL
- Output paths

Limitations

- You cannot use environment parameters in recipes.
- You cannot use environment parameters in plans.
- Environment parameter names must be unique within the project or workspace. You can apply override values to them at runtime.
- You cannot use environment parameters in Deployment Manager. For more information, see *Overview of Deployment Manager*.

Edit Environment Parameter Value

Administrators can change the default value for an environment parameter.

NOTE: Modifying the default value of an environment parameter immediately applies the change across the entire environment. All subsequent job and plan runs are affected.

NOTE: After you have created an environment parameter, you cannot change the name. You must create a new environment parameter.

Steps:

1. In the Environment Parameters page, locate the parameter whose default value you wish to modify.
2. In the More menu, select **Edit value**.
3. Enter a new value, and click **Save**.

Delete Environment Parameter

When you delete an environment parameter, all references to the parameter are converted to empty string values. Job executions can fail, and recipe steps can break.

Tip: If you delete an environment parameter and then recreate it using the exact same name, references to the parameter are updated with the new default value, which replaces the empty string value for the deleted parameter.

Steps:

1. In the Environment Parameters page, locate the parameter whose default value you wish to modify.
2. In the More menu, select **Delete**.

Export Environment Parameters

You can export the environment parameters from your project or workspace.

NOTE: All environment parameters are exported at the same time into a ZIP file. Do not modify this file outside of the Designer Cloud application .

Steps:

- 1. In the Environment Parameters page, select **More menu > Export**.
- 2. The ZIP file is downloaded to your local desktop.

Import Environment Parameters

If you have exported a set of environment parameters, you can import them into another workspace or project.

NOTE: If an environment parameter that you are importing has a name that conflicts with an environment parameter that already exists, you must either rename the imported parameter or delete it from the import set.

Steps:

- 1. In the Environment Parameters page, select **More menu > Import**.
- 2. Select or drag-and-drop the ZIP file. Click **Import**.

NOTE: Select the ZIP file or its embedded JSON5 file for import.

- 3. The Import environment parameters dialog is displayed:

Import environment parameters

Name

env. region

Default value

01

Name

env. bucket_name

Default value

myBucket

Name

env. name

Default value

Prod

Name

env. admin_contact

Default value

admin@example.com

Add another

Cancel

Save

Figure: Import environment parameters dialog

- 4. Review each environment variable and its assigned value from the import package:
 - a. Modify values as needed.
 - b. To delete a parameter from the import process, click the Trash icon.
 - c. To add another parameter as part of the import package, click **Add another**.
- 5. To save your changes and complete the import, click **Save**.
- 6. The environment parameters and your modifications to them are imported.

System Services and Logs

Contents:

- *Download Logs*
 - *Support logs*
 - *Log directory*
 - *Artifact Storage Service*
 - *Authorization Service*
 - *Batch Job Runner*
 - *Configuration Service*
 - *Connector Configuration Service*
 - *Conversion Service*
 - *Data Service*
 - *Java UDF Service*
 - *Java VFS Service*
 - *Job Metadata Service*
 - *Machine Learning Service*
 - *Migration*
 - *Nginx Service*
 - *Optimizer Service*
 - *Scheduling Service*
 - *Spark Job Service*
 - *Supervisord Server*
 - *Time-Based Trigger Service*
 - *VFS Service*
 - *Webapp Service*
 - *Additional logs*
 - *Job logs*
-

The Designer Cloud powered by Trifacta® platform provides the following major services. For each of the listed service, any relevant logs are listed.

The logging levels for many of these services can be modified through the Admin Settings page. See *Configure Logging for Services*.

Download Logs

Support logs

For support use, the most meaningful logs and configuration files can be downloaded from the application. Select **Resources menu > Download logs**.

NOTE: If you are submitting an issue to *Alteryx Support*, please download these files through the application.

For more information, see *Download Logs Dialog*. The admin version of this dialog enables downloading logs by timeframe, job ID, or session ID. For more information, see *Admin Download Logs Dialog*.

Log directory

System logs are maintained in the following directory: `/opt/trifacta/logs`

Trifacta® administrators can access the logs through the Designer Cloud application . Use the following URL:

<hostname>:<port_number>/logs

Available logs

Filename	Description
jobgroups/	Directory of logs for transformation jobs by Id. <div>Tip: If you are troubleshooting a failed job, please acquire the job logs from the Job Details page when you contact <i>Alteryx Support</i>. See <i>Job Details Page</i>.</div>
jobs/	Directory of logs for other kinds of jobs, such as sampling or ingest, by Id.
nginx/	Temporary storage for nginx server. No log files are stored here.
artifact-storage-service.access.log	Access logs for the artifact storage service.
artifact-storage-service.log	Application logs for the artifact storage service.
authorization-service.access.log	Access logs for the authorization service.
authorization-service.log	Application logs for the authorization service.
batch-job-runner.access.log	Access logs for the batch job runner service. Batch job runner service manages transformation jobs and scheduling. More information is below.
batch-job-runner.job-status.log	Status information on batch job runner jobs.
batch-job-runner.log	Application logs for the batch job runner service.
configuration-service.access.log	Access logs for the configuration service service. Configuration service is used for managing configuration that can be changed at runtime for different workspaces and users. Some of these settings are available through the Designer Cloud application . See <i>Workspace Settings Page</i> .
configuration-service.log	Application logs for the configuration service.
connector-configuration-service.log	Application logs for the connector configuration service.
connector-configuration-service.access.log	Access logs for the connector configuration service.
conversion-service.access.log	Access logs for the conversion service. Conversion service is used for converting from various inputs formats and to various output formats.
conversion-service.log	Application logs for the conversion service.
data-service.access.log	Access logs for the data service. Data service is used for interacting with relational sources. More information is below.

data-service.log	Application logs for the data service.
java-vfs-service.access.log	Access logs for the Java VFS service.
java-vfs-service.log	Application logs for the Java VFS service.
job-metadata-service.access.log	Access logs for the job metadata service.
job-metadata-service.log	Application logs for the job metadata service.
migration.log	Application logs for database migrations performed for the webapp service.
ml_service.access.log	Access logs for the ml (machine-learning) service. Machine learning service is used for predictive interaction, suggestion ranking, pattern profiling, pattern suggestions, and collecting user action logs. More information is below.
ml_service.log	Application logs for the ml (machine-learning) service.
nginx_service.log	Application logs for the nginx service. More information is below.
optimizer-service.access.log	Access logs for the optimizer service.
optimizer-service.log	Application logs for the optimizer service.
protobuf-events.log	Client events around column values, user selections, and recipe editing.
proxy_access.log	Access logs for the nginx server.
proxy_error.log	Error logs for the nginx server.
scheduling-service.access.log	Access logs for the scheduling service. Scheduling service is used for scheduling jobs at a specific time. More information is below.
scheduling-service.log	Application logs for the scheduling service.
secure-token-service.access.log	Access logs for the secure token service. secure-token-service is used for securely storing tokens for some external services, such as Azure AD and Databricks and OAuth2-connected datastores.
secure-token-service.log	Application logs for the secure token service
spark-job-service.log	Application logs for the Spark job service. Spark job service is used for interfacing with cluster-based Spark service to plan and execute Spark jobs. More information is below.
supervisord.log	Logs for the supervisord system service. supervisord manages the starting, stopping, and restarting of services for the Designer Cloud powered by Trifacta platform . More information is below.
time-based-trigger-service.access.log	Access logs for the time-based trigger service. Time-based trigger service is used for managing the triggers for scheduled jobs. More information is below.
time-based-trigger-service.log	Application logs for the time-based trigger service.

<code>vfs-service.access.log</code>	Access logs for the VFS service. VFS service is used for managing loading of files from various supported datastores. More information is below.
<code>vfs-service.log</code>	Application logs for the VFS service.
<code>webapp.access.log</code>	Access logs for the Webapp service. Webapp service serves the Designer Cloud application to users. More information is below.
<code>webapp.log</code>	Application logs for the Webapp service. More information is below.
<code>webapp.sql-error.log</code>	Error log for SQL issued from the Webapp service.
<code>webworker.log</code>	Event logs for webworkers running in browser clients. Webworkers run in the background of a browser client's Trifacta session and are used for predictive interaction, suggestion ranking, pattern profiling, pattern suggestions, and collecting user action logs.

Artifact Storage Service

Description: Manages storage of feature-specific usage data, such as value mappings.

Log File	Can Help With
<code>artifact-storage-service-access.log</code>	Access issues to the service.
<code>artifact-storage-service.log</code>	Transactions with the database for the features that use it.

Authorization Service

Description: Manages access permissions for workspace objects.

Log File	Can Help With
<code>authorization-service-access.log</code>	Access issues to the service.
<code>authorization-storage-service.log</code>	Transactions with the database for the features that use it.

Batch Job Runner

Description: This service manages the tracking of jobs submitted to the backend running environment.

Log File	Can Help With
<code>batch-job-runner.log</code>	<ul style="list-style-type: none"> • Service errors and crashes • Determine execution environment of the job. Search for: <ul style="list-style-type: none"> • <code>LocalJobRunner</code> = local execution in Photon • <code>YARNRunner</code> = execution in Spark • Communication errors back from environment • Status information on jobs • Status information on counts of job retries

For more information on this service, see *Configure Batch Job Runner*.

Configuration Service

Description: Service for managing configurations at the user, workspace, and system levels, which can be changed at runtime.

Log File	Can Help With
configuration-service-access.log	Issues accessing the service.
configuration-service.log	Configuration problems.

Some configuration service options are surfaced in the Designer Cloud application . See *Workspace Settings Page*.

Connector Configuration Service

Description: Service for managing metadata for connector types.

Log File	Can Help With
connector-configuration-service-access.log	Issues accessing the service.
connector-configuration-service.log	Configuration problems.

Conversion Service

Description: Service converts some formats for input or output. For example, Microsoft Excel workbooks must be ingested through the conversion service and stored as separate CSVs for each worksheet.

Log File	Can Help With
conversion-service-access.log	Issues accessing the service.
conversion-service.log	Problems with ingest jobs for datasets whose sources must be converted.

Data Service

Description: Service prepares queries against JDBC interfaces, using internal REST API calls.

Log File	Can Help With
data-service.log	<ul style="list-style-type: none">Initialization of communications through JDBC interfaceQuery failures

For more information on this service, see *Configure Data Service*.

Java UDF Service

Description: Service enables the execution of Java-based user-defined functions within a transform recipe.

Log File	Can Help With
java-udf-service.log	<ul style="list-style-type: none">Status of the service <div>Tip: You can pass through messages on errors through Logger to this log, which can assist in diagnosing issues.</div>

For more information, see *User-Defined Functions*.

Java VFS Service

Description: Service manages metadata associated with jobs during execution.

Log File	Can Help With
java-vfs-service.log	<ul style="list-style-type: none">• Help with issues accessing data on ADLS Gen2• Status of the service

For more information, see *Configure Java VFS Service*.

Job Metadata Service

Description: Service manages metadata associated with jobs during execution.

Log File	Can Help With
job-metadata-service.log	<ul style="list-style-type: none">• Help with job phases and status• Status of the service

Machine Learning Service

Description: ML service provides machine learning capabilities for the platform.

Log File	Can Help With
ml_service.log	<ul style="list-style-type: none">• This log is likely to contain information that is only useful if the ML service has crashed.

Migration

Description: Migration applies to database migrations executed as part of upgrading the platform.

Log File	Can Help With
migration.log	<ul style="list-style-type: none">• Connectivity errors and other issues that may have occurred during migration.

Nginx Service

Description: Nginx is a proxy server embedded in the platform that serves the web application and other resources.

Log File	Can Help With
nginx_service.log	<ul style="list-style-type: none">• This log may be useful in identifying any warnings that occurred when the nginx services starts.• The <code>nginx</code> server may contain log information for the server that provides HTTP access.

Optimizer Service

Description: Handles optimizations of the queries for data from relational sources.

Log File	Can Help With
optimizer-service.log	<ul style="list-style-type: none">• Service-related issues
optimizer-service.access.log	<ul style="list-style-type: none">• Gives information about accessed routes

Proxy

Description: The proxy (nginx) service manages requests from the user interface to the other components of the platform.

Log File	Can Help With
proxy_access.log	<ul style="list-style-type: none">• Shows any requests made through the nginx to the port used by the Designer Cloud powered by Trifacta platform .
proxy_error.log	<ul style="list-style-type: none">• Contains any errors thrown by the nginx service when a request is made to the port used by the Designer Cloud powered by Trifacta platform .

Scheduling Service

Description: Handles all metadata related to scheduling.

Log File	Can Help With
scheduling-service.log	<ul style="list-style-type: none">• Schedule-related issues
scheduling-service.access.log	<ul style="list-style-type: none">• Gives information about accessed routes

Spark Job Service

Description: Service that manages jobs processed on Spark.

Log File	Can Help With
spark-job-service.log	<ul style="list-style-type: none">• Status of the service• Debugging issues with Spark jobs• See <i>Configure for Spark</i>.

Supervisord Server

Description: Process that starts, stops, and restarts services in the platform.

--	--

Log File	Can Help With
<code>supervisord.log</code>	<ul style="list-style-type: none"> Status information from platform services

Time-Based Trigger Service

Description: Handles all metadata related to the trigger service

Log File	Can Help With
<code>time-based-trigger-service.log</code>	<ul style="list-style-type: none"> schedules not triggering correctly
<code>time-based-trigger-service.access.log</code>	<ul style="list-style-type: none"> Gives information about accessed routes

VFS Service

Description: Loads data from the various filesystems supported by the platform, both in the front-end user interface and in batch mode when the Trifacta Photon running environment is enabled. For more information, see *Running Environment Options*.

Log File	Can Help With
<code>vfs-service.log</code>	<ul style="list-style-type: none"> Client connection issues Status issues with backend components Information on batch jobs that cannot be started

Webapp Service

Description: Loads data from the various filesystems supported by the platform in the front-end user interface.

Log File	Can Help With
<code>webapp.log</code>	<ul style="list-style-type: none"> Client connection issues Status issues with backend components Information on batch jobs that cannot be started
<code>webapp.access.log</code>	<ul style="list-style-type: none"> Provides access information to the Webapp service and routes.

Additional logs

Job logs

The following sources of information may provide information related to job status and performance:

- job log
- spark log
- cdf script
- yarn application logs
(if log aggregation is enabled)
- platform configuration file
(`trifacta-conf.json`)
- batch job runner log

- spark service log
- hadoop conf directory

Storage Maintenance

Contents:

- *Trifacta Storage*
 - *Service logs*
 - *Job logs*
 - *Base Storage Layer*
 - *Temp files*
 - *Samples and profile statistics*
 - *Datasets*
 - *Storage for features*
-

This page provides some tips and guidelines for maintaining your backend storage.

NOTE: Except for temporary files that it creates as part of normal operations or storage used as part of feature execution, Designer Cloud Powered by Trifacta® Enterprise Edition does not remove files from the backend storage for safety reasons. Unless resources have been provided to you by Alteryx, management of the backend datastore is the responsibility of the customer.

NOTE: Designer Cloud Powered by Trifacta Enterprise Edition does not store data on the Trifacta node where the software is installed.

NOTE: Designer Cloud Powered by Trifacta Enterprise Edition does not modify source data.

Trifacta Storage

Log files are stored by default in the following location on the Trifacta node:

```
/opt/trifacta/logs
```

Service logs

Service log files are automatically auto-rotated at 50 MB. For more information on configuring log rotation, see *Configure Logging for Services*.

Job logs

Logs related to job execution are not automatically rotated.

NOTE: Job log files can accumulate over time. As a good rule of thumb, you can set up a recurring job through an external scheduler to purge old job logs that are older than six months.

Job log files are stored in the following directories:

```
/opt/trifacta/logs/jobs  
/opt/trifacta/logs/jobgroups
```

They are organized by job identifier in sub-directories.

For more information on job logs, see *Diagnose Failed Jobs*.

Base Storage Layer

Temp files

Job temp files

Temporary files may be written to the temporary directory on the backend datastore, particularly during job execution.

```
/tmp
```

NOTE: These files may be purged during restarts of the platform.

Spark temp files

During execution of jobs, Spark may use the following directories on backend storage for storage of temporary files:

```
/user/<UserID>  
/trifacta/tempfiles
```

Samples and profile statistics

The Designer Cloud powered by Trifacta platform generates your samples and profiling statistics in one of the following directories for each user:

- The default directory:

```
/trifacta/queryResults/.trifacta
```

- The user-defined output directory

NOTE: These files should be removed on a periodic basis.

Datasets

While samples and job results may be retained on backend storage, the Designer Cloud powered by Trifacta platform does not store your source data.

NOTE: Datasets removed from the Library are removed as references to the product. The underlying data is not actually deleted.

Storage for features

The following features do store data on the base storage layer.

File conversion

Data sources that are stored in a binary format, such as PDF or Excel, or that require additional processing, such as JSON, must be converted to file format that can be natively ingested by the Designer Cloud powered by Trifacta platform . Typically these files are stored in the base storage layer in CSV format.

This feature is enabled by default.

JDBC ingestion

When JDBC ingestion is enabled, some objects used in sampling that are sourced from JDBC sources may be stored in the base storage layer for faster retrieval. After job execution, these objects are deleted, or if datasource caching is enabled, are moved to the appropriate datasource cache.

For more information, see *Configure JDBC Ingestion*.

Datasource caching

If datasource caching has been enabled, cached objects can be stored in either a global or user-specific cache. For more information, see *Configure Data Source Caching*.

Backup and Recovery

Contents:

- *Stop All Services*
 - *Perform manual backups*
 - *Restart*
 - *Recovery*
-

This section provides overview information on the key data and metadata that should be managed by your enterprise backup and recovery policies.

NOTE: This section covers how to perform a basic cold backup of the product. Hot backups are not supported.

All backups should be performed in accordance with your enterprise's backup and recovery policies.

Stop All Services

Before you begin, the Designer Cloud powered by Trifacta platform and databases should be stopped. See *Start and Stop the Platform*.

Perform manual backups

Back up platform files

The following directories on the Trifacta node should be backed up on a regular basis:

Configuration files:

You can back up all key configuration files into the `/tmp` directory using the following commands:

```
cp -R /opt/trifacta/conf /tmp/conf
cp /etc/init.d/trifacta /tmp/trifacta.service
cp -R /opt/trifacta/pkg3p/tripache/conf/conf.d /tmp/conf.d
cp -R /opt/trifacta/services/data-service/build/conf/vendor /tmp/vendor
cp -R /opt/trifacta/hadoop-deps /tmp/trifacta-hadoop-deps
```

License file:

You should back up your license key:

```
cp /opt/trifacta/license/license.json /tmp/license.json
```

See *License Key*.

Log files:

Optionally, you can choose to back up your log files:

Tip: Designer Cloud powered by Trifacta platform upgrades may be faster if the log directory is empty. Before you upgrade, you may wish to back up this directory, empty it, and then restore your backup after the upgrade.

```
cp -R /opt/trifacta/logs /tmp/logs
```

Back up databases

The Designer Cloud powered by Trifacta platform utilizes the following databases as part of normal operations. These databases should be backed up on a regular basis:

Database Name	Databaseld	Description
Main DB	trifacta	Stores users and metadata for flows, including datasets, and recipes.
Jobs DB	trifacta-activiti	Stores and maintains job execution status and details.
Scheduling DB	trifactascheduling-service	Stores metadata for scheduled jobs.
Time-based Trigger DB	trifactatimebasedtriggerservice	Additional database required for scheduled jobs.
Configuration Service DB	trifactaconfiguration-service	Stores configuration settings for the workspace.
Artifact Storage Service DB	trifactaartifactstorage-service	Stores feature usage data such value mappings for the standardization feature.
Job Metadata Service DB	trifactajobmetadata-service	Stores metadata on job execution.
Authorization Service DB	trifactaauthorization-service	Storage of object permissions.
Orchestration Service DB	trifactaorchestration-service	Storage of plans, triggers, tasks, and snapshots.
Optimizer Service DB	trifactaoptimizer-service	Storage of SQL queries for optimization during job execution.
Secure Token Service DB	trifactasecuretoken-service	Storage of STS tokens for use in accessing third-party systems.
Connector Configuration Service DB	trifactaconnectorconfiguration-service	Storage of metadata information on connector types.

For more information on setting up these databases, see *Install Databases*.

Location of db tools - PostgreSQL

Depending on your operating system, you can find the backup tools in the following location.

CentOS/RHEL - PostgreSQL 12:

NOTE: These locations apply to PostgreSQL 12.

```
/usr/pgsql-12/bin/pg_dump  
/usr/pgsql-12/bin/psql
```

Ubuntu:

```
/usr/lib/postgresql/9.6/bin/pg_dump
/usr/lib/postgresql/9.6/bin/psql
```

Location of db tools - MySQL

Please locate the following programs in your MySQL distribution:

```
mysqldump
mysql
```

Manual backup commands

The following commands can be used to back up the databases.

PostgreSQL

For more information on command options, see <https://www.postgresql.org/docs/9.6/static/backup.html>.

NOTE: These commands must be executed as the `trifacta` user.

NOTE: The following commands are for PostgreSQL 12.3 for all supported operating systems. For specific commands for other versions, please see the database documentation.

Tip: You may see performance improvements by backing up and restoring using .TAR files. However, there is a risk that .TAR support could change in the future. For more information, please see the PostgreSQL documentation.

Trifacta DB:

NOTE: If you are providing a dump of the `trifacta` database to *Alteryx Support*, please include a dump of the `trifactaauthorizationservice` database, as well.

```
pg_dump trifacta > trif_tridB_bkp_<date>.sql
```

Jobs DB:

```
pg_dump trifacta-activiti > trif_actDB_bkp_<date>.sql
```

Scheduling DB:

```
pg_dump trifactaschedulingservice > trif_schDB_bkup_<date>.sql
```

Time-Based Trigger DB:


```
pg_dump trifactatimebasedtriggerservice > trif_tbttsDB_bkup_<date>.sql
```

Configuration Service DB:

```
pg_dump trifactaconfigurationservice > trif_confservDB_bkup_<date>.sql
```

Artifact Storage DB:

```
pg_dump trifactaartifactstorageservice > trif_artifactstorageservDB_bkup_<date>.sql
```

Job Metadata Service DB:

```
pg_dump trifactajobmetadataservice > trif_jobmetadataservDB_bkup_<date>.sql
```

Authorization Service DB:

```
pg_dump trifactaauthorizationservice > trif_authorizationservDB_bkup_<date>.sql
```

Orchestration Service DB:

```
pg_dump trifactaorchestrationservice > trif_orchestrationservDB_bkup_<date>.sql
```

Optimizer Service DB:

```
pg_dump trifactaoptimizerservice > trif_optimizerservDB_bkup_<date>.sql
```

Secure Token Service DB:

```
pg_dump trifactasecuretokenservice > trif_securetokenservDB_bkup_<date>.sql
```

Connector Configuration Service DB:

```
pg_dump trifactaconnectorconfigurationservice > trif_connectorconfigurationservDB_bkup_<date>.sql
```

MySQL

For more information on command options, see
<https://dev.mysql.com/doc/refman/5.7/en/mysqldump-sql-format.html>.

```
su - mysql
```

NOTE: The following commands are for MySQL 5.7 for all supported operating systems. For specific commands for other versions, please see the database documentation.

Trifacta DB:

NOTE: If you are providing a dump of the trifacta database to *Alteryx Support*, please include a dump of the trifactaauthorizationservice database, as well.

```
mysqldump trifacta > trif_triDB_bkp_<date>.sql
```

Jobs DB:

```
mysqldump trifacta-activiti > trif_actDB_bkp_<date>.sql
```

Scheduling DB:

```
mysqldump trifactaschedulingservice > trif_schDB_bkup_<date>.sql
```

Time-Based Trigger DB:

```
mysqldump trifactatimebasedtriggerservice > trif_tbtsDB_bkup_<date>.sql
```

Configuration Service DB:

```
mysqldump trifactaconfigurationservice > trif_confservDB_bkup_<date>.sql
```

Artifact Storage DB:

```
mysqldump trifactaartifactstorageservice > trif_artifactstorageservDB_bkup_<date>.sql
```

Job Metadata Service DB:

```
mysqldump trifactajobmetadataservice > trif_jobmetadataservDB_bkup_<date>.sql
```

Authorization Service DB:

```
mysqldump trifactaauthorizationservice > trif_authorizationservDB_bkup_<date>.sql
```

Orchestration Service DB:

```
mysqldump trifactaorchestrationservice > trif_orchestrationservDB_bkup_<date>.sql
```

Optimizer Service DB:

```
mysqldump trifactaoptimizerservice > trif_optimizerservDB_bkup_<date>.sql
```

Secure Token Service DB:

```
mysqldump trifactasecuretokenservice > trif_securetokenservDB_bkup_<date>.sql
```

Connector Configuration Service DB:

```
mysqldump trifactaconnectorconfigurationservice > trif_connectorconfigurationservDB_bkup_<date>.sql
```

Scheduling

You can schedule nightly execution of these backups using a third-party scheduler such as cron.

Restart

You can restart the Designer Cloud powered by Trifacta platform now. See *Start and Stop the Platform*.

Recovery

See *Platform Rollback*.

Platform Rollback

Contents:

- *Prerequisites*
 - *Rollback Overview*
 - *Stop services*
 - *Create new versions of databases - PostgreSQL*
 - *Uninstall and reinstall Trifacta software*
 - *Restore databases and config files*
 - *Restart*
 - *Verify*
-

In the event that an upgrade or hotfix to your instance of the Designer Cloud powered by Trifacta® platform has run into issues that cannot be repaired in the upgraded instance, you can follow the steps in this section to rollback to your previous version.

NOTE: Before you perform a rollback, you should review the set of issues with Alteryx first. For more information, please contact *Alteryx Support*.

Prerequisites

In order to complete the rollback in a timely manner, please verify that you have access to the following:

Access

You must:

- Acquire root user access to the Trifacta node.
- Acquire database access to uninstall and reinstall the Trifacta databases.

Tip: You should communicate to any affected users the required maintenance and expected outage window.

Backups

If you do not have the following, you cannot perform a rollback. These items cannot be acquired from Alteryx.

- Backups of your pre-upgrade Trifacta configuration files
- Backups of your pre-upgrade Trifacta databases

The following can be acquired from Alteryx if you do not have them:

- RPM installers for the previous version. If any Hotfixes have been applied to the previous version, you should acquire and use the latest Hotfix RPM for your re-install.
- PDF documentation for the previous version.

Rollback Overview

To recover the Designer Cloud powered by Trifacta platform based on backups, please complete the following sections.

NOTE: When the databases are restored, internal identifiers such as job IDs, are reset in an order that may not correspond to the expected order. Consequently, references to specific identifiers may be corrupted. After restoring the databases, you should clear the job logs.

NOTE: If any of the hosts, pathnames, or credentials have changed since the backups were performed, these updates must be applied through `trifacta-conf.json` or through the Admin Settings page after the restoration is complete.

Stop services

You must stop the Designer Cloud powered by Trifacta platform .

Steps:

1. Login to the Trifacta node as root user.
2. Stop the Trifacta service:

```
service trifacta stop
```

Create new versions of databases - PostgreSQL

Before you restore, you must drop each database and create new versions of the databases that you wish to restore.

NOTE: The following assumes that the roles for each database are already created.

Login as a user that can run admin commands for PostgreSQL. This user may vary between deployments.

Trifacta database:

```
psql -c "DROP DATABASE trifacta;"
psql -c "CREATE DATABASE trifacta WITH OWNER trifacta;"
```

Jobs database:

NOTE: Please note the escaped quotes in the `CREATE DATABASE` command for this database.

```
psql -c "DROP DATABASE \"trifacta-activiti\";"
psql -c "CREATE DATABASE \"trifacta-activiti\" WITH OWNER trifactaactiviti;"
```

Scheduling database:

```
psql -c "DROP DATABASE trifactaschedulingservice;"
psql -c "CREATE DATABASE trifactaschedulingservice WITH OWNER trifactascheduling;"
```

Time-based Trigger Service database:

```
psql -c "DROP DATABASE trifactatimebasedtriggerservice;"
psql -c "CREATE DATABASE trifactatimebasedtriggerservice WITH OWNER trifactatimebasedtriggerservice;"
```

Configuration Service database:

(Release 6.0 and later)

```
psql -c "DROP DATABASE trifactaconfigurationservice;"
psql -c "CREATE DATABASE trifactaconfigurationservice WITH OWNER trifactaconfigurationservice;"
```

Artifact Storage Service database:

(Release 6.0 and later)

```
psql -c "DROP DATABASE trifactaartifactstorageservice;"
psql -c "CREATE DATABASE trifactaartifactstorageservice WITH OWNER trifactaartifactstorageservice;"
```

Job Metadata Service database:

(Release 6.4 and later)

```
psql -c "DROP DATABASE trifactajobmetadataservice;"
psql -c "CREATE DATABASE trifactajobmetadataservice WITH OWNER trifactajobmetadataservice;"
```

Authorization Service database:

(Release 7.1 and later)

```
psql -c "DROP DATABASE trifactaauthorizationservice;"
psql -c "CREATE DATABASE trifactaauthorizationservice WITH OWNER trifactaauthorizationservice;"
```

Orchestration Service database:

(Release 7.1 and later)

```
psql -c "DROP DATABASE trifactaorchestrationservice;"
psql -c "CREATE DATABASE trifactaorchestrationservice WITH OWNER trifactaorchestrationservice;"
```

Optimizer Service database:

(Release 7.6 and later)

```
psql -c "DROP DATABASE trifactaoptimizerservice;"
psql -c "CREATE DATABASE trifactaoptimizerservice WITH OWNER trifactaoptimizerservice;"
```

Secure Token Service database:

(Release 8.1 and later)

```
psql -c "DROP DATABASE trifactasecuretokenservice;"
psql -c "CREATE DATABASE trifactasecuretokenservice WITH OWNER trifactasecuretokenservice;"
```

Connector Configuration Service database:

(Release 8.1 and later)

```
psql -c "DROP DATABASE trifactaconnectorconfigurationservice;"
psql -c "CREATE DATABASE trifactaconnectorconfigurationservice WITH OWNER
trifactaconnectorconfigurationservice;"
```

Uninstall and reinstall Trifacta software

Steps:

1. Uninstall the current version of the Trifacta software:

NOTE: All platform and cluster configuration files are preserved. User metadata is preserved in the Trifacta database.

CentOS/RHEL:

```
sudo rpm -e trifacta
```

Ubuntu:

```
sudo apt-get remove trifacta
```

2. Perform a clean install of the Trifacta software provided in your distribution. See *Install*.

Restore databases and config files

You can use the following commands in the sections for manual restores of:

- Databases
 - PostgreSQL
 - MySQL
- Configuration files

Restore databases - PostgreSQL

Trifacta database:

```
psql --dbname=trifacta < trif_triDB_bkp_<date>.sql
```

Jobs database:

NOTE: Please note the escaped quotes in the `CREATE DATABASE` command for this database.

```
psql --dbname="trifacta-activiti" < trif_actDB_bkp_<date>.sql
```

Scheduling database:

```
psql --dbname=trifactaschedulingservice < trif_schDB_bkup_<date>.sql
```

Time-based Trigger Service database:

```
psql --dbname=trifactatimebasedtriggerservice < trif_tbtsDB_bkup_<date>.sql
```

Configuration Service database:

(Release 6.0 and later)

```
psql --dbname=trifactaconfigurationservice < trif_confservDB_bkup_<date>.sql
```

Artifact Storage Service database:

(Release 6.0 and later)

```
psql --dbname=trifactaartifactstorageservice < trif_artifactstorageservDB_bkup_<date>.sql
```

Job Metadata Service database:

(Release 6.4 and later)

```
psql --dbname=trifactajobmetadataservice < trif_jobmetadataservDB_bkup_<date>.sql
```

Authorization Service database:

(Release 7.1 and later)

```
psql --dbname=trifactaauthorizationsservice < trif_authorizationservDB_bkup_<date>.sql
```

Orchestration Service database:

(Release 7.1 and later)

```
psql --dbname=trifactaorchestrationservice < trif_orchestrationservDB_bkup_<date>.sql
```

Optimizer Service database:

(Release 7.6 and later)

```
psql --dbname=trifactaoptimizerservice < trif_optimizerservDB_bkup_<date>.sql
```

Secure Token Service database:

(Release 8.1 and later)

```
psql --dbname=trifactasecuretokenservice < trif_securetokenservDB_bkup_<date>.sql
```

Connector Configuration Service database:

(Release 8.1 and later)

```
psql --dbname=trifactaconnectorconfigurationservice < trif_connectorconfigurationservDB_bkup_<date>.sql
```

Restore databases - MySQL

For details, see <https://dev.mysql.com/doc/refman/5.7/en/reloading-sql-format-dumps.html>.

Login:

```
su - mysql
```

Trifacta database:

```
mysql trifacta < trif_triDB_bkp_<date>.sql
```

Jobs database:

```
mysql trifacta-activiti < trif_actDB_bkp_<date>.sql
```

(Release 4.1 and later) Scheduling database:

```
mysql trifactaschedulingservice < trif_schDB_bkup_<date>.sql
```

(Release 4.1 and later) Time-based Trigger Service database:

```
mysql trifactatimebasedtriggerservice < trif_tbtsDB_bkup_<date>.sql
```

(Release 6.0 and later) Configuration Service database:

```
mysql trifactaconfigurationsservice < trif_confservDB_bkup_<date>.sql
```

(Release 6.0 and later) Artifact Storage Service database:

```
mysql trifactaartifactstorageservice < trif_artifactstorageservDB_bkup_<date>.sql
```

(Release 6.4 and later) Job Metadata Service database:

```
mysql trifactajobmetadataservice < trif_jobmetadataservDB_bkup_<date>.sql
```

(Release 7.1 and later) Authorization Service database:

```
mysql trifactaauthorizationservice < trif_authorizationservDB_bkup_<date>.sql
```

(Release 7.1 and later) Orchestration Service database:

```
mysql trifactaorchestrationservice < trif_orchestrationservDB_bkup_<date>.sql
```

(Release 7.6 and later) Optimizer Service database:

```
mysql trifactaoptimizerservice < trif_optimizerservDB_bkup_<date>.sql
```

(Release 8.1 and later) Secure Token Service database:

```
mysql trifactasecuretokenservice < trif_securetokenservDB_bkup_<date>.sql
```

(Release 8.1 and later) Connector Service database:

```
mysql trifactaconnectorconfigurationservice < trif_connectorconfigurationservDB_bkup_<date>.sql
```

Restore config files

Restore your configuration files. The following commands assume that they were backed up to the `/tmp` directory on the node:

```
cp /tmp/trifacta-conf.json /opt/trifacta/conf/trifacta-conf.json
cp /tmp/env.sh /opt/trifacta/conf/env.sh
cp /tmp/trifacta.service /etc/init.d/trifacta
```

Restart

Apply any patches or maintenance updates that may have been provided to you.

Restart the platform. See *Start and Stop the Platform*.

Verify

Login and verify operations. See *Verify Operations*.

Admin Reference

This section describes the pages where admin users can modify settings and users for the Designer Cloud powered by Trifacta® platform . Most of these pages are available through the Admin console in the Designer Cloud application . Additional admin reference materials are included.

Deployment Manager Page

Contents:

- *Access*
- *Deployment Hierarchy*
- *Deployments View*
- *Releases View*
- *Flows View*

Through the Deployment Manager page, you interact with flows that you have imported into your Production instance of the Designer Cloud powered by Trifacta® platform . Through this interface, you can activate Production versions of your flows or rollback to previous versions as needed.

NOTE: The Deployment Manager is available only in a Production environment, which is a special instance of the Designer Cloud powered by Trifacta platform designed to support production use of your flows. For more information, see *Overview of Deployment Manager*.

Access

A Production environment can be accessed in either of the following ways:

- You are given access to a separate instance of the Designer Cloud powered by Trifacta platform configured for Production use only.
- On any instance, the Deployment role is added to your user account by a Trifacta administrator.

For more information, see *Configure Deployment Manager*.

Deployment Hierarchy

In a Production environment, a **deployment** is version-managed flow and all of its dependencies, including other dependent flows. Through the Deployment Manager, an individual deployment is structured in the following hierarchy:

Hierarchy Level	Object	Description
1	deployment	When you open the Deployment Manager, you can review all of the deployments that have been created in the environment. A deployment is container for releases.
2	release	<p>When you select a deployment, you can explore its releases. A release is an individual instance of an imported flow and its dependencies (an import package). Each time that the import package is re-imported into the Production instance, a new release is created and made the active release for the deployment.</p> <p>You can activate previous releases as needed through the context menus in Deployment Manager.</p>
3	flow or flows	<p>Within a release, you can explore the flows that were included in the import package for the release:</p> <ul style="list-style-type: none">• The primary flow is the one that is executed when:<ul style="list-style-type: none">• Its release is the active one for the deployment• The job for the deployment is executed• Any secondary flows are the flows on which the primary flow depends for data.<ul style="list-style-type: none">• During export from the source instance, all objects in secondary flows are included in the package. There may be objects in a secondary flow that are unused in the Production instance.

Deployments View

When you open the Deployment Manager, you can explore all of the deployments in the Production instance.

Deployments Create		<input type="text" value="Search name..."/>	
Name	Releases	Last Updated By	Last Updated At
Deployment 02	0 Releases	SteveO	2019-03-15 10:36:57
Deployment 01	2 Releases	SteveO	2019-03-15 10:27:54

Figure: Deployment Manager

To create a new deployment, click **Create**.

1. Enter a name for the deployment, and click **Create**.
2. To create a new release, click the created deployment. See Releases View below.

Actions:

- **Search:** Enter values in the search textbox to search deployment names. Matching occurs in real-time.
- **Edit name:** You can change the name of your deployment as needed.
- **Delete:** Select this option to remove the deployment, all of its releases, and all of the flows within each release.

You cannot undo deleting a deployment. Any results generated from jobs run for the deployment are not removed from the output location and are still accessible through the Job History page of the Production instance.

Releases View

Through Releases view, you can import new packages to create new releases and activate them, roll back to previous releases, and remove releases that are no longer in use.

Deployments

Deployment 01

Import Package

Search Package ID

Release	Package ID	Notes	Created At	Updated At	
<div><div></div>3</div>	0fec54a0	USDA Farmers Market	2019-03-15 10:27:54	2019-03-15 10:27:54	<div><div></div>Active</div>
<div><div></div>1</div>	0fec54a0	USDA Farmers Market	2019-03-15 10:21:20	2019-03-15 10:27:28	

Figure: Releases View

To create a new release, click **Import Package**.

NOTE: Before you import a package, you must apply any import mapping rules to the deployment. These rules map values and objects in the package to corresponding values in the new instance. For more information, see *API Task - Define Deployment Import Mappings*.

1. Navigate your local environment to select the ZIP file containing the flow and its dependencies from the source instance.
2. Click **Import**.
3. The release is added to Releases view.

For more information on importing, see *Import Flow*.

Actions:

Action	Description
Search	Enter values in the search textbox to search package identifiers. Matching occurs in real-time.
Activate	Make the selected release the active one for the deployment. When jobs are executed at the deployment level, the primary flow for the active release is executed.
Export	Export the release for use in another instance of the platform. See <i>Export Flow</i> .
Delete	<div>Delete the release from the Production instance.</div> <div>NOTE: If the release was imported from a Dev instance on the same platform, the Dev instance of the release is not removed.</div> <div>NOTE: Deletion of a release does not remove any results generated from it. Those results are still accessible through the Job History page.</div>

Flows View

When you open a release, you can review the flows contained in it. You can explore the flow or flows that were included in the import package for the selected release.

Deployments > Deployment 01 0fec54a0 (Release 3)			Search name...
Name	Datasets	Updated At ▾	
 USDA Farmers Market 2014 Flow	2 Datasets	Today at 10:27 AM	

Figure: Flows View

Actions:

- **Search:** Enter values in the search textbox to search flow names. Matching occurs in real-time.
- Click a flow name to explore the flow in Flow View.

NOTE: Avoid making changes to a flow in a Production instance. You can run ad-hoc jobs, but you should avoid making changes to the objects or their structure through Flow View in Production instances. Scheduling should be done through the command line.

For more information, see *Flow View Page*.

Admin Console

Contents:

- *Users*
 - *Roles*
 - *Workspace settings*
 - *Admin settings*
 - *AWS settings*
 - *Environment parameters*
 - *OAuth 2.0 clients*
-

Through the Admin console, admin users can modify settings and users at the system and workspace level, as well as run health checks and manage the license for the Designer Cloud powered by Trifacta® platform . Select **User menu > Admin console**.

NOTE: You must be an administrator to access this feature.

Users

Invite, disable, and remove users. Change roles, as needed. For more information, see *Users Page*.

Roles

Create roles and assign permissions to them for access to objects created in the Designer Cloud application . For more information, see *Roles Page*.

Workspace settings

Review and edit settings applicable to the workspace. For more information, see *Workspace Settings Page*.

Admin settings

NOTE: The Admin Settings page is only available to administrators.

Platform configuration settings:

- Review and manage configuration for the Designer Cloud powered by Trifacta platform .
- Configure settings for external services.
- Manage user accounts.
- Diagnostics and license management
- Platform restart

For more information. see *Admin Settings Page*.

AWS settings

If per-user access to AWS has been enabled, individual users must apply personal access credentials to their account to gain access to resources on S3 through AWS. For more information, see *AWS Settings Page*.

Environment parameters

Define parameters that apply to the entire environment and are available for use by all users. For more information, see *Environment Parameters Page*.


OAuth 2.0 clients

Administrators can create and manage clients for accessing an OAuth 2.0 app in an external platform such as a relational datastore.

NOTE: Before you create an OAuth 2.0 client, you must have created an OAuth 2.0 app in the target system, to which your client can connect. For more information, see *Enable OAuth 2.0 Authentication*.

For more information, see *OAuth 2.0 Clients Page*.

Users Page

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

The Users page enables adding, disabling, or removing users from your project or workspace. You can also reset passwords and change roles.

Users

<

1

2

>

Test User

AllEnabledDisabled

Name	Email	Status	Last log in
<div>TU</div> Test User 1786321687		Enabled	Today at 7:41 AM
<div>TU</div> Test User 2292989979		Enabled	Today at 7:32 AM
<div>TU</div> Test User19933547		Enabled	Today at 12:02 PM
<div>TU</div> Test User21197568		Enabled	Today at 11:31 AM
<div>TU</div> Test User22395996		Enabled	Today at 11:39 AM
<div>TU</div> Test User29277976		Enabled	Today at 12:01 PM
<div>TU</div> Test User33101513		Enabled	Today at 11:59 AM
<div>TU</div> Test User46556047		Enabled	Today at 12:00 PM
<div>TU</div> Test User57276476		Enabled	Today at 12:04 PM
<div>TU</div> Test User57741240		Enabled	Today at 11:37 AM

Figure: Users Page

Tabs:

- Click one of the tabs to display all users or a filtered list based on user status.

Fields:

- Name:** Display name for the user. Click the name of the user to review details about the user account. See *User Details Page*.
- Email:** Username (email address of users)
- Status:** Current status of the user. See "Status" below.
- Last login:** Timestamp for the last time that the user logged in to the Designer Cloud application

Actions:

- Search:** Enter text to begin searching for specific usernames or email addresses.

Context menu actions:

For each user, you can perform the following actions in the context menu:

- **Configure storage:** If per-user access is enabled, you can configure the access credentials for individual users, either using key-secret combinations or IAM roles. For more information, see *Configure Your Access to S3*.
- **Edit:** Modify user properties, including platform roles. See "Edit Users" below.
- **Reset password:** Self-service password reset is enabled by default. If enabled, click this option to send an email to the user to reset his or her password.

NOTE: Only platform administrators can reset a user's password. Workspace admins cannot.

Disable: When a user is disabled, the user cannot access the Designer Cloud application .

- The disabled user still counts against the project or workspace limit.
- All of the user's assets are retained.

NOTE: Schedules owned by a disabled user continue to execute. An admin can disable the schedule. See *Schedules Page*.

- Assets that are owned by the user become inaccessible to other users that have access.
- To permit access again, select **Enable**.

Status

Users can be set to one of the following statuses:

- **Enabled:** User can log in and use the Designer Cloud application normally.
- **Disabled:** User account has been disabled by an administrator. User cannot use the project or workspace.

NOTE: A disabled user's assets are still stored within the Designer Cloud application . However, the user cannot access them. Ownership of these objects has not been transferred. An administrator has ownership privileges on the user's objects.

Edit Users

To modify a user account, please complete the following steps.

NOTE: For security reasons, an administrator is not permitted to edit some settings in the administrator's own account.

Steps:

1. Locate the user in the list of users.
2. In the context menu on the right side of the user's listing, select **Edit**.
3. In the Edit User dialog, modify the following properties as needed:

Name: The display name of the user.

Email: The email address is used as the login identifier. This value cannot be modified.

Roles: Select or remove the roles to assign to the user. For more information, see *Roles Page*.

SSO Principal: If SSO is enabled, set this value to be the SSO principal value associated with this user.

NOTE: Required value for each user if SSO is enabled. See *Configure SSO for AD-LDAP*.

Hadoop Principal: If secure impersonation is enabled, set this value to be the Hadoop principal value associated with this user.

NOTE: The user principal value should not include the realm.

NOTE: Hadoop principal is a required value if secure impersonation is enabled. See *Configure for Secure Impersonation*.

NOTE: If Kerberos is enabled, verify that all user principals that use the platform are also members of the group of the keytab user.

Deployment management: When selected, this user is assigned the deployment role in the platform. In a Development environment, this role can be added to a user's account to enable access to the Deployment Manager.

NOTE: Deployment management user accounts are intended for managing production execution of flows. These users have a different and limited user interface in the Designer Cloud application . There should be a limited number of these accounts.

NOTE: Only platform administrators can assign the Deployment management role. Workspace admins cannot.

Tip: A deployment user should be assigned the flow author role. Lesser flow roles may prevent the deployment user from properly importing and managing flows. See *Roles Page*.

- In a Production environment where the Deployment Manager applies to the entire instance, this role does not apply.
- For more information, see *Configure Deployment Manager*.
- For more information on Deployment Manager, see *Overview of Deployment Manager*.

Platform admin: When selected, the user is granted admin privileges over the platform. These privileges include user administration, ability to modify platform settings, and permissions to use admin-only API endpoints.


NOTE: Avoid providing the Platform admin permission to a large number of users.

To save your changes, click **Edit user**.

User Details Page

Contents:


- Overview Tab
 - Group membership
 - Roles
 - Privileges
 - User Details
- Assets Tab
 - Bulk transfer via API

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Review details about the selected user's account and make changes, including transferring ownership of all of the user's assets.

Overview Tab

Users

 ana

Edit

...

Overview

Assets

Roles

Default

Workspace admin

Privileges

This user is an admin which allows them to access all objects within the workspace and configure workspace settings.

User details

Status

Enabled

Last log in

Today at 4:30 PM

Created on

10/14/2020

Figure: Overview tab

Context menu actions:

For each user, you can perform the following actions in the context menu:

- **Edit:** Modify user properties, including platform roles. See *Users Page*.
- **Configure storage:** If per-user access is enabled for the workspace, you can configure the access credentials for individual users, either using key-secret combinations or IAM roles. For more information, see *Configure Your Access to S3*.
- **Reset password:** Self-service password reset is enabled by default. If enabled, click this option to send an email to the user to reset his or her password.

NOTE: Only platform administrators can reset a user's password. Workspace admins cannot.

- **Disable:** When a user is disabled, the user cannot access the Designer Cloud application .

- The disabled user still counts against the workspace limit.
- All of the user's assets are retained.
- Assets that are owned by the user become inaccessible to other users that have access.

NOTE: Schedules owned by a disabled user continue to execute. An admin can disable the schedule. See *Schedules Page*.

- To permit access again, select **Enable**.
- **Remove:** When a user is removed, the user's account and all assets are removed from the Designer Cloud application .
 - At time of removal, you can choose to transfer ownership of the user's assets to another user. If those assets are not transferred, they are removed with the user's account.

If you must recover a removed user or that user's assets, please contact Alteryx Support within 14 days of the deletion.

- An admin cannot remove himself or herself from the project or workspace.

Group membership

Any group assignments are listed in this section.

NOTE: This feature is in Beta release.

For more information on groups, see *Configure Users and Groups*.

Roles

The roles assigned to the user are listed. For more information, see *Roles Page*.

Privileges

In this section, you can review the maximal set of privileges that are assigned to the user.

- Privileges are additive.
- For more information, see *Privileges and Roles Reference*.

User Details

Information on the current status and recent activity of the user. If the user has any platform roles, they are listed here. These roles can be enabled or disabled when you edit the user.

Assets Tab

In the Assets tab, you can review all assets owned by the user by asset category.

Asset transfer cannot be undone.

Tip: After asset ownership has been transferred, the original user retains access to the asset or assets as a collaborator, except for macros.

Users

ana

ana

Edit ...

Overview

Assets

All assets (140)

Transfer all

Search assets 1

Flows (16)

Connections (7)

Plans (4)

Datasets (112)

Macros (1)

☐ Flow

☐ Untitled Flow - 32

☐ validation_test

☐ Untitled

☐ Untitled Flow - 31

☐ Untitled Flow - 30

Last updated ▾

06/07/2022

06/07/2022

06/07/2022

05/24/2022

03/17/2022

Figure: Assets tab

To transfer one or more assets:

1. Click the asset category on the left side.
2. Locate or search for the asset.

Tip: Search results are constrained to the currently selected category.

3. Click the checkbox next to the asset or assets that you wish to transfer.
4. You can select assets from other categories, too.
5. Then, click **Transfer**.
6. Specify the user to whom you wish to transfer the assets. Then, click **Transfer ownership**.

NOTE: Asset transfer cannot be undone.

To transfer all of a user's assets:

1. Click **Transfer all**.
2. Specify the user to whom you wish to transfer the assets. Then, click **Transfer ownership**.


NOTE: Asset transfer cannot be undone.

For more information, see *Transfer Asset Ownership*.

Bulk transfer via API

Administrators can also perform transfers of ownership of all of a user's assets using APIs.
<https://api.trifacta.com/ee/es.t/index.html#operation/transferUserAssetsInCurrentWorkspace>

Roles Page

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the Roles page, an admin can create roles and assign one or more of them to Trifacta users.

- A **role** is a set of privileges that can be assigned to one or more Trifacta users.
 - A **privilege** is a level of access to a type of user-generated objects.
 - For more information on these terms, see *Overview of Authorization*.
- For more information on managing roles, see *Create Role*.

You can also apply roles to groups that are synched from your enterprise LDAP provider. For more information, see *Configure Users and Groups*.

Roles

Create role




Name	Privileges	Last Updated
 default	Connection author, Plan author, Flow author	Last Sunday at 3:21 PM
 viewer_connections	Connection viewer	Today at 10:38 AM
 viewer_flows	Flow viewer	Today at 10:38 AM

Figure: Roles Page

The list of current roles is displayed in the Roles page. To create a new role, click **Create role**. See *Create Role Dialog*.

Columns:

- **Name:** The name of the role must be unique within the project or workspace.
- **Privileges:** The comma-separated list of privileges associated with the role. When a user is assigned the role, these privileges are available to the user.

Tip: Hover over the entry in the Privileges column to see additional detail on the privileges assigned to this role.

- **Last Updated:** Timestamp of when the role was most recently updated.

Context menu:

On the right side of the screen, you can select from a context menu for each available role.

- **Edit:** Modify the role. See *Create Role Dialog*.

NOTE: You cannot modify the admin role.

NOTE: All new and existing users are assigned the default role. Changes to this role may affect all existing users and any users that are invited in the future.

- **Assign role:** Assign the role to users.


NOTE: When you assign or un-assign a role, the privileges are immediately applied to the assigned user's account. The user does not need to re-login to see the changes.

- You can un-assign a role from users through the Role Details page. Select the role, and then click the Users tab. For more information, see *Role Details Page*.
- **Delete:** Delete the role.

You are permitted to delete roles that are currently assigned to users. Deleting a role may remove privileges from one or more users. This action cannot be undone. Before deleting, you should verify the list of users assigned to the role. For more information, see *Role Details Page*.

NOTE: You cannot delete the `default` or admin roles.

Create Role Dialog

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.


To create a new role that you can assign to users, click **Create role** in the Roles page.

Create role


Name

viewer_flows


Privileges

 Flows

Viewer

 Connections

None

 Plans

None

Cancel

Save

Figure: Create Role Dialog

In this dialog, you assign one or more privileges to the defined role.

Name: Enter a name for your role. This name must be unique within the roles in the current project or workspace.

Privileges:

Tip: You can create a role with no privileges, which may be useful for disabling access to objects without disabling the account itself. In this case, all other roles would need to be removed from the assigned user.

- **Flows:** These privileges govern the actions that users can perform on flows.
- **Connections:** These privileges govern the actions that users can perform on connections.
- **Plans:** These privileges govern the actions that users can perform on plans.
- For more information, see *Privileges and Roles Reference*.

To finish creating your role, click **Save**.

This role is now available for assigning to users. See *Roles Page*.

Role Details Page

Contents:

- *Overview tab*
- *Users tab*
- *Groups tab*



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the Role Details page, you can review the privileges assigned to the role and assign the role to Trifacta users.

Actions:

The following actions are available.

- **Edit:** Modify the role.

NOTE: All new and existing users are assigned the `default` role. Changes to this role may affect all existing users and any users that are added in the future.

NOTE: You cannot edit the admin role.

See *Create Role Dialog*.

- **Assign role:** Assign the role to one or more users.

NOTE: When you assign or un-assign a role, the privileges are immediately applied to the assigned user's account. The user does not need to re-login to see the changes.

- **Delete:** Delete the role.

You are permitted to delete roles that are currently assigned to users. Deleting a role may remove privileges from one or more users. This action cannot be undone. Before deleting, you should verify the list of users assigned to the role in the Users tab.

NOTE: You cannot delete the `default` or admin roles.

Overview tab

In the Overview tab, you can review the privileges for the role and the current number of users that have been assigned the role.

Roles
Flow-Author

Edit
...

Overview
Users

Privileges

Flow author
View, execute, create, edit, delete, and share flows

Role Details

Users	1
Created On	07/02/2020
Last Updated	07/02/2020

Figure: Role Details Page - Overview tab

For more information on the listed privileges, see *Privileges and Roles Reference*.

Roles can be created through the Roles page. For more information, see *Roles Page*.

Users tab

In the Users tab, you can review the list of users who have been assigned the role.

Roles
Flow-Author

Edit
...

Overview
Users

Name	Email
Testuser 1234	testuser1234@trifacta.com

Figure: Role Details Page - Users tab

Columns:

- Name:** Display name of the user.

NOTE: You cannot modify the Name value for the default role.

- Email:** Email address for the user, which is used to login to the Designer Cloud application .

Context menu:

On the right side of the screen, you can select the following options from the context menu for each user:

- Unassign from role:** Select this option to remove the role from the user.

Groups tab

In the Groups tab, you can review the list of groups to which the role has been assigned.

NOTE: Optionally, users of the platform can be synched with your enterprise LDAP service provider. For more information, see *Configure Users and Groups*.

Workspace Settings Page

Contents:

- *General*
 - *Filter Job History*
 - *Hide underlying file system to users*
 - *Locale*
 - *Self service password reset*
 - *Session Tags: Enable the use of session tags when assuming an IAM role*
 - *Session Tags: The name of the session tag that holds the username as its value*
 - *Session duration*
 - *Show file location*
 - *Storage directories*
 - *User messaging*
- *API*
 - *API Access Token*
 - *Allow users to generate access tokens*
 - *Maximum lifetime for user generated access tokens (days)*
- *Connectivity*
 - *Connectivity feature*
 - *Custom SQL query*
 - *Enable S3 connectivity*
 - *Enable conversion of standard JSON files via conversion service*
 - *Max endpoints per JDBC REST connection*
 - *Upgrade to connectivity*
 - *Upgrade to upload large files*
- *Flows, recipes and plans*
 - *Collaborative suggestions*
 - *Column from examples*
 - *Editor Scheduling*
 - *Export*
 - *Import*
 - *Maximum number of files to read in a directory for the initial sample*
 - *Plan feature*
 - *Sample downloads*
 - *Schematized output*
 - *Sharing*
 - *UI for range join*
 - *Webhooks*
- *Job execution*
 - *Combine Spark Transform and Profile jobs into one.*
 - *Custom Spark Options Feature*
 - *Databricks Cluster Policies*
 - *Databricks Job Management*
 - *Databricks Job Runs Submit Fallback*
 - *Ignore publishing warnings for running jobs*
 - *Logical and physical optimization of jobs*
 - *SQL Scripts*
 - *Schema validation feature*
 - *Schema validation option in job settings*
 - *Schema validation option to fail job*
 - *Spark Whitelist Properties*
 - *Trifacta Photon execution*
- *Scheduling and parameterization*
 - *Include Hidden Files in Parameterization*
 - *Parameterization*
 - *Schedule list*

- *Scheduling feature*
- *Publishing*
 - *Avro output format*
 - *CSV output format*
 - *Hyper output format*
 - *JSON output format*
 - *Parquet output format*
 - *Publish job results*
 - *Publishing actions options*
- *Notifications*
 - *Email notification feature*
 - *Email notification trigger when flow jobs fail*
 - *Email notification trigger when flow jobs succeed*
 - *Email notification trigger when plans run*
 - *Sharing email notifications*
- *Experimental features*
 - *Cache data in the Transformer intelligently*
 - *Default language*
 - *Edit recipes without loading sample*
 - *Enable/Disable data grid from view options*
 - *Execution time threshold (in milliseconds) to control caching in the Transformer*
 - *Language localization*
 - *Show user language preference*
 - *Wrangle to Python Conversion*

The following settings can be customized for the user experience in your workspace. When you modify a setting, the change is immediately applied to the workspace. To access the page, select **User menu > Admin console > Workspace settings**.

NOTE: Users may not experience the changed environment until each user refreshes the application page or logs out and in again.

Enablement Options:

NOTE: Any values specified in the Workspace Settings page applies exclusively to the specific workspace and override any system-level defaults.

Option	Description
Default	<p>The default value is applied. This value may be inherited from higher-level configuration.</p> <div> Tip: You can review the default value as part of the help text. </div>
Enabled	<p>The setting is enabled.</p> <div> NOTE: If the setting applies to a feature, the feature is enabled. Additional configuration may be required. See below. </div>
Disabled	The setting is disabled.
Edit	Click Edit to enter a specific value for the setting.

General

Filter Job History

Set the default number of days of jobs that are displayed in the Job History page. Default value is 180 days.

Tip: You can filter the dates of the jobs displayed in the Job History page.

For more information, see *Job History Page*.

Hide underlying file system to users

When enabled, workspace users cannot see locations in the default storage layer.

Locale

Set the locale to use for inferring or validating data in the application, such as numeric values or dates. The default is `United States`.

NOTE: After saving changes to your locale, refresh your page. Subsequent executions of the data inference service use the new locale settings.

For more information, see *Locale Settings*.

Self service password reset

When enabled, workspace users can reset their own passwords via link on the login page.

Session Tags: Enable the use of session tags when assuming an IAM role

If you are using IAM roles to request temporary credentials for access to AWS resources, you can enable the use of session tags to make those requests. When a **session tag** is submitted, the Trifacta user is provided access to AWS resources based on the user's corresponding permissions within AWS, instead of having to specify those permissions in the Designer Cloud powered by Trifacta platform . This method leverages the existing permission infrastructure in your enterprise and simplifies the use of IAM roles in the Designer Cloud application .

NOTE: After enabling the use of session tags, you must spin up a new EMR cluster, which forces EMR to use the newly deployed credential provider JAR file.

NOTE: Additional configuration is required. For more information, see *Configure AWS Per-User Auth for Temporary Credentials* .

Session Tags: The name of the session tag that holds the username as its value

When Session Tags: Enable the use of session tags when assuming an IAM role is enabled, you must specify the name of the session tag to be submitted to AWS containing the username of the Trifacta user requesting resources. Default value is `trifacta-user`.

For more information, see *Configure AWS Per-User Auth for Temporary Credentials*.

Session duration

Specify the length of time in minutes before a session expires. Default is 10080 (one week).

Show file location

When enabled, workspace users can see the locations of source and output files within the application.

Storage directories

Allow members of the workspace to change paths to their upload and output results locations through their user profile.

For more information, see *Storage Config Page*.

User messaging

When enabled, workspace users can explore content through the Designer Cloud application .

API

API Access Token

When accessing the REST APIs, you can optionally use a token for simpler use and enhanced security.

NOTE: This feature may not be available in all environments.

NOTE: API access tokens must be enabled to use the API reference documentation available through the User menu.

For more information, see *Access Tokens Page*.

Allow users to generate access tokens

When enabled, individual workspace users can generate their own personal access tokens, which enable access to REST APIs. For more information, see *Manage API Access Tokens*.

Maximum lifetime for user generated access tokens (days)

Defines the maximum number of days that a user-generated access token is permitted for use in the product.

Tip: To permit generation of access tokens that never expire, set this value to -1.

For more information, see *Manage API Access Tokens*.

Connectivity

Connectivity feature

When enabled, workspace users can create connections to relational datasources.

NOTE: Disabling this feature hides existing relational connections.

See *Relational Access*.

Custom SQL query

When enabled, users can create custom SQL queries to import datasets from relational tables. For more information, see *Enable Custom SQL Query*.

Enable S3 connectivity

When enabled, base connectivity to S3 is enabled for workspace users.

NOTE: Additional platform configuration is required. See *S3 Access*.

Enable conversion of standard JSON files via conversion service

When enabled, the Designer Cloud application utilizes the conversion service to ingest JSON files and convert them to a tabular format that is easier to import into the application.

NOTE: This feature is enabled by default but can be disabled as needed. The conversion process performs cleanup and re-organization of the ingested data for display in tabular format.

When disabled, the Designer Cloud application uses the old version of JSON import, which does not restructure the data and may require additional recipe steps to manually structure it into tabular format.

NOTE: The legacy version of JSON import is required if you are working with compressed JSON files or only Newline JSON files.

NOTE: Although imported datasets and recipes created under v1 of the JSON importer continue to work without interruption, the v1 version is likely to be deprecated in a future release. You should switch your old imported datasets and recipes to using the new version. Instructions to migrate are provided at the link below.

Max endpoints per JDBC REST connection

For a REST API connection to a JDBC source, this parameter defines the maximum number of endpoints that can be defined to use the connection.

Avoid modifying this value unless you are experiencing timeouts or failures to connect.

For more information, see *REST API Connections*.

Upgrade to connectivity

When enabled, workspace users are presented with the option to upgrade to a plan that supports connection to external data sources, if the feature is current disabled.

Upgrade to upload large files

When enabled, workspace users are presented with the option to upgrade to a plan that supports uploading large files, if the feature is current disabled.

Flows, recipes and plans

Collaborative suggestions

If desired, you can enable the inclusion of suggestion cards that are generated from recent use of the Designer Cloud application . As the application gathers more information about how you or members of your workspace apply transformations to your data, the suggestions become more meaningful for the data that you are processing.

NOTE: No data is shared with Alteryx or any system outside of the Designer Cloud powered by Trifacta platform .

These collaborative suggestion cards can be generated from individual usage or from workspace level usage. These suggestions appear under the Recently Used heading in the side panel.

NOTE: This feature requires the machine learning service, which is enabled by default. For more information, see *Miscellaneous Configuration*.

When this feature is enabled, individual users can still choose to opt-out of sharing their usage data with this feature. See *User Profile Page*.

Option	Description
disabled	Collaborative suggestions are not surfaced in the application.
personal	Collaborative suggestions are based on the individual user's previous transformations.
workspace	Collaborative suggestions are based on the transformations from all users in the workspace.
Default	The default setting for the workspace is applied.

Column from examples

When enabled, users can access a tool through the column menus that enables creation of new columns based on example mappings from the selected column.

Editor Scheduling

When enabled, flow editors are also permitted to create and edit schedules. For more information, see *Flow View Page*.

NOTE: The Scheduling feature may need to be enabled in your environment. When enabled, flow owners can always create and edit schedules.

When this feature is enabled, plan collaborators are also permitted to create and edit schedules. For more information, see *Plan View Page*.

Export

When enabled, workspace users are permitted to export their flows and plans. Exported flows can be imported into other workspaces or product editions.

NOTE: If plans are been enabled in your workspace, enabling this flag applies to flows and plans.

- For more information, see *Export Flow*.
- For more information, see *Export Plan*.

Import

When enabled, workspace users are permitted to import exported flows and plans.

NOTE: If plans have been enabled in your workspace, enabling this flag applies to flows and plans.

- For more information, see *Import Flow*.
- For more information, see *Import Plan*.

Maximum number of files to read in a directory for the initial sample

When the Designer Cloud application is generating an initial sample of data for your dataset from a set of source files, you can define the maximum number of files in a directory from which the sample is generated. This limit is applied to reduce the overhead of reading in a new file, which improves performance in the Transformer page.

Tip: The initial sample type for files is generated by reading one file after another from the source. If the source is multiple files or a directory, this limit caps the maximum number of files that can be scanned for sampling purposes.

NOTE: If the files in the directory are small, the initial sample may contain the maximum number of files and less than the maximum size permitted for a sample. You may see fewer rows than expected.

If the generated sample is unsatisfactory, you can generate a new sample using a different method. In that case, this limit no longer applies. For more information, see *Overview of Sampling*.

Plan feature



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When enabled, users can create plans to execute sequences of recipes across one or more flows. For more information, see *Plans Page*.

For more information on plans and orchestration, see *Overview of Operationalization*.

Sample downloads

When enabled, members can download the contents of the Transformer page at any time. For an individual step, a member can download the current sample, as modified by the current recipe up to the point of the current step. For more information, see *Recipe Panel*.

Schematized output

When enabled, all output columns for all types of outputs are typecast to their annotated types. This feature is enabled by default.

For non-schematized outputs, the Designer Cloud powered by Trifacta platform enforces casting of all values to the annotated data type of the column by default. For example, if the output value is `-3.4` and the data type for the output column is Integer, the platform enforces Integer type casting and writes a null value instead.

- `true`: All output values must match the data type of the output columns, or a null value is written.

- `false`: All output values are written in their output form, regardless of the column's data type.

Sharing

When enabled, workspace users are permitted to share flows and plans with other users in the workspace.

NOTE: If plans have been enabled in your workspace, enabling this flag applies to flows and plans.

- For more information, see *Share a Flow*.
- For more information, see *Share a Plan*.

UI for range join

When enabled, workspace users can specify join key matching across a range of values. For more information, see *Configure Range Join*.

Webhooks



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When enabled, webhook notification tasks can be configured on a per-flow basis in Flow View page. Webhook notifications allow you to deliver messages to third-party applications based on the success or failure of your job executions. For more information, see *Create Flow Webhook Task*.

NOTE: Additional configuration may be required. See *Configure Webhooks*.

Job execution

Combine Spark Transform and Profile jobs into one.

When enabled, the transform and profiling tasks of a job executed on the Spark running environment are combined. The profiling task is executed as a part of the transform task, which eliminates any time spent orchestrating the profiling task and accessing the profiler input file on storage.

NOTE: When these two tasks are combined, publishing actions are not undertaken if the profiling task fails.

For more information on these actions, see *Configure for Spark*.

In the Job Details page, combined jobs appear in a Transform with profile card. See *Job Details Page*.

Custom Spark Options Feature

When enabled, users can override Spark configuration options for output objects before running Spark jobs.

Tip: When enabled, a default set of Spark configuration options is available for users. Additional properties can be specified through the Spark Whitelist Properties setting.

See *Enable Spark Job Overrides*.

Databricks Cluster Policies

When enabled, this feature allows the Designer Cloud powered by Trifacta platform to leverage cluster policies that you have created for use when creating new Databricks clusters for job execution.

NOTE: You must create cluster policies before enabling this feature. Each user may select a cluster policy to use. Additional configuration and considerations may be required. If no policy is selected, jobs may still be executed.

For more information:

- *Configure for Azure Databricks*
- *Configure for AWS Databricks*

Databricks Job Management

Enables job execution on Databricks through a secondary method. When enabled, Databricks jobs are executed via the run/submit API endpoint, which avoids the job quota limitation imposed by Databricks clusters. This flag also enables deletion of Databricks jobs from the Databricks workspace.

For more information on these options:

- *Configure for Azure Databricks*
- *Configure for AWS Databricks*

Databricks Job Runs Submit Fallback

When this flag is enabled, users can execute Databricks jobs using the runs/submit API method as a fallback when the job quota limit is reached for a Databricks workspace.

For more information:

- *Configure for Azure Databricks*
- *Configure for AWS Databricks*

Ignore publishing warnings for running jobs

When enabled, a user may execute a job if the previously saved location is not available for the current IAM permissions used to run the job. Default is `Enabled`.

Tip: Setting this value to `Enabled` is helpful for resolving changes in IAM permissions.

When disabled, the **Run Job** button is disabled if the previously saved location is not available through IAM permissions.

Tip: Setting this value to `Disabled` prevents execution of jobs that are going to fail at publication time, which can be expensive in terms of time and compute costs.

Logical and physical optimization of jobs

When enabled, the Designer Cloud application attempts to optimize job execution through logical optimizations of your recipe and physical optimizations of your recipes interactions with data.

NOTE: This feature requires the optimizer service, which is enabled by default, and the optimizer service database, which is installed by default. For more information on installing the database, see *Install Databases*.

This workspace setting can be overridden for individual flows. For more information, see *Flow Optimization Settings Dialog*.

SQL Scripts

When enabled, users may define SQL scripts to execute as part of a job's run. Scripts can be executed before data ingestion, after output publication, or both through any write-supported relational connection to which the user has access.

For more information, see *Create Output SQL Scripts*.

Schema validation feature

When enabled, by default the structure and ordering of columns in your import datasets are checked for changes before data is ingested for job execution.

Tip: Schema validation can be overridden for individual jobs when the schema validation option is enabled in the job settings. See below.

Errors are immediately reported in the Job Details page. See *Job Details Page*.

For more information on schema validation, see *Overview of Schema Management*.

Schema validation option in job settings

When the schema validation feature and this setting are enabled, users can make choices on how individual jobs are managed when schema changes are detected. This setting is enabled by default.

For more information on schema validation, see *Overview of Schema Management*.

Schema validation option to fail job

When schema validation is enabled, this setting specifies the default behavior when schema changes are found.

- When enabled, jobs are failed when schema changes are found, and error messages are surfaced in the Designer Cloud application .
- When disabled, jobs are permitted to continue.
 - Jobs may ultimately fail due to schema changes.
 - Jobs may result in bad data being written in outputs.
 - Job failures may be more challenging to debug.

Tip: Setting this value to `Disabled` matches the behavior of the Designer Cloud application from before schema validation was possible.

Tip: This setting can be overridden for individual jobs, even if it is disabled.

Errors are immediately reported in the Job Details page. See *Job Details Page*.

For more information on schema validation, see *Overview of Schema Management*.

Spark Whitelist Properties

Comma-separated list of additional Spark properties to be whitelisted for configuration of output objects while running Spark jobs.

NOTE: The Custom Spark Options feature must be enabled.

See *Enable Spark Job Overrides*.

Trifacta Photon execution

When enabled, users can choose to execute their jobs on Trifacta Photon, a proprietary running environment built for execution of small- to medium-sized jobs in memory on the Trifacta node.

Tip: When enabled, you can select to run jobs on Photon through the Run Job page. The default running environment is the one that is best for the size of your job.

When Trifacta Photon is disabled:

- You cannot run jobs on the local running environment. All jobs must be executed on a clustered running environment.
- Trifacta Photon is used for Quick Scan sampling jobs. If Trifacta Photon is disabled, the Designer Cloud application attempts to run the Quick Scan job on another available running environment. If that job fails or no suitable running environment is available, the Quick Scan sampling job fails.

Scheduling and parameterization

Include Hidden Files in Parameterization

When enabled, hidden files and hidden directories can be searched for matches for wildcard- or pattern-based parameters when importing datasets.

Tip: This can be useful for importing data from generated profiles, which are stored in the `.profiler` folder in a job output directory.

NOTE: Scanning hidden folders may impact performance. For existing imported datasets with parameters, you should enable the inclusion of hidden folders on individual datasets and run a test job to evaluate impact.

For more information, see *Parameterize Files for Import*.

Parameterization

By default, the Designer Cloud powered by Trifacta platform supports the application of parameters to imported datasets. Datetime, wildcard, or variable parameters can be used to operationalize execution of jobs on different versions of the same dataset.

When enabled, users can create parameters, which can be applied to import, creating sample, and outputs. For more information, see *Overview of Parameterization*.

Schedule list

When enabled, administrators and workspace administrators can see a list of all schedules in the workspace.

Scheduling feature

When enabled, workspace users can schedule the execution of flows. See *Add Schedule Dialog*.

Publishing

Avro output format

When enabled, members can generate outputs in Avro format.

CSV output format

When enabled, members can generate outputs in CSV format.

Hyper output format



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

When enabled, members can generate outputs in Hyper format for publication and use on Tableau Server.

JSON output format

When enabled, members can generate outputs in JSON format.

Parquet output format

When enabled, members can generate outputs in Parquet format.

Publish job results

When enabled, workspace users are permitted to publish results through the Output Destinations tab in the Job Details page to external datastores.

NOTE: These external datastores must be enabled and configured. See *Connection Types*.

For more information, see *Job Details Page*.

Publishing actions options

When enabled, users are permitted to create custom publishing actions for their jobs.

When disabled, users must accept the default publishing actions.

Notifications

Email notification feature

When enabled, the Designer Cloud powered by Trifacta platform can send email notifications to users based on the success or failure of jobs. By default, this feature is `Enabled`.

NOTE: This feature requires access to an SMTP server to send emails. For more information, see [Enable SMTP Email Server Integration](#).

Email notification trigger when flow jobs fail

When email notifications are enabled, you can configure the default setting for the types of failed jobs that generate an email to interested stakeholders. The value set here is the default value for each flow in the workspace.

Settings:

Setting	Description
Default (any jobs)	By default, email notifications are sent on failure of any job.
Never send	Email notifications are never sent for job failures.
Scheduled jobs	Notifications are sent only when scheduled jobs fail.
Manual jobs	Notifications are sent only when ad-hoc (manually executed) jobs fail. <div>Tip: Jobs executed via API are Ad hoc jobs.</div>
Any	Notifications are sent for all job failures.

Individual users can opt out of receiving notifications or configure a different email address. See [Email Notifications Page](#).

Emailed stakeholders are configured by individual flow. For more information, see [Manage Flow Notifications Dialog](#).

Email notification trigger when flow jobs succeed

When email notifications are enabled, you can configure the default setting for the types of successful jobs that generate an email to interested stakeholders. The value set here is the default value for each flow in the workspace.

For more information on the settings, see the previous section. Default setting is `Default (any jobs)`.

Individual users can opt out of receiving notifications or configure a different email address. See [Email Notifications Page](#).

Emailed stakeholders are configured by individual flow. For more information, see [Manage Flow Notifications Dialog](#).

Email notification trigger when plans run

You can configure the default trigger for email notifications when a plan runs. Default setting is `Default (all runs)`.

Setting	Description
Default (all runs)	By default, email notifications are sent to users for all plan runs.
All runs	Emails are sent for all runs.
Failed runs	Emails are sent for failed runs only.
Success runs	Emails are sent for successful runs only.

Sharing email notifications

When email notifications are enabled, users automatically receive notifications whenever an owner shares the plan or flow with the user.

Individual users can opt out of receiving notifications. For more information, see *Preferences Page*.

Experimental features

These experimental features are not supported.

Experimental features are in active development. Their functionality may change from release to release, and they may be removed from the product at any time. Do not use experimental features in a production environment.

These settings may or may not change application behavior.

Cache data in the Transformer intelligently

NOTE: This feature is in Beta release.

When enabled, this feature allows the Designer Cloud application to cache data from the Transformer page periodically based on Trifacta Photon execution time. This feature enables users to move faster between recipe steps.

Default language

Select the default language to use in the Designer Cloud application .

Edit recipes without loading sample

When enabled, you can perform edits in the Transformer page without loading a sample in the data grid.

Tip: This feature can be helpful when you know the edits that need to be performed and do not need sample data to perform the corrections. You can also use it to switch the active sample without loading.

In Flow View, select **Edit recipe without datagrid** from the context menu on the right side when the recipe is selected.

Enable/Disable data grid from view options

When enabled, you can enable or disable live previewing in the data grid of the Transformer page. Disabling can improve performance. These options are available in the **Show/hide data grid options** drop-down in the status bar at the bottom of the Transformer page:

- Edit with data grid
 - When the data grid is disabled, you may not be able to edit some recipe steps. For steps that you can edit, select Preview to see the effects of the step on the data. When you select Preview, the data grid is re-enabled.
- Show column histogram
 - When the data grid is enabled, you can choose to disable the column histograms in the data grid, which can improve performance.

For more information, see *Data Grid Panel*.

Execution time threshold (in milliseconds) to control caching in the Transformer

NOTE: This feature is in Beta release.

When intelligent caching in the Transformer is enabled, you can set the threshold time in milliseconds for when Trifacta Photon updates the cache. At each threshold of execution time in Trifacta Photon, the output of the intermediate recipe (CDF) steps are cached in-memory, which speeds up movements between recipe steps in the Designer Cloud application .

Language localization

When enabled, the Designer Cloud application is permitted to display text in the selected language.

Show user language preference

When enabled, individual users can select a preferred language in which to display text in the Designer Cloud application .

NOTE: This experimental feature requires installation of a language resource file on the Trifacta node. For this release, only U.S. English (default) and Korean are supported. For more information, please contact *Alteryx Support*.

Users can make personal language selections through their preferences. See *Account Settings Page*.

Wrangle to Python Conversion

NOTE: As of Release 9.7, Wrangle to Python conversion has been deprecated. For more information, please see *End of Life and Deprecated Features*.



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Alpha feature: When enabled, you can use an API endpoint to generate Python Pandas code that completes the steps required to generate an output in Python.

This feature may be modified or removed in a future release without warning. It is intended for demonstration purposes only and should not be enabled in a production environment.

For more information, see *API Task - Wrangle Output to Python*.

Tip: You can download and install the Python SDK to integrate use of the Designer Cloud application in your Python environment. Use the visual tools of the Designer Cloud application to build your transformations, and then generate Python Pandas code for use in your Python data pipelines. For more information, see *Python SDK*.

Admin Settings Page

Contents:

- Platform Settings
 - External Service Settings
 - AWS EMR Cluster ID
 - AWS Region
 - Resource Bucket
 - Resource Path
 - Services
 - View Logs
 - Tricheck
 - SMTP settings
 - Upload License
 - Restart
-

Admin users of the Designer Cloud powered by Trifacta® platform can change settings through the Designer Cloud application. Login as an admin user. Select **User menu > Admin Console > Admin Settings**.

NOTE: You must be an administrator to access this feature.

Platform Settings

Do not modify settings through the Admin Settings page and through `trifacta-conf.json` at the same time. Saving changes in one interface wipes out any unsaved changes in the other interface. Each requires a platform restart to apply the changes.

Platform administrators can change any parameter value that is available through the web application. Enter some or all of parameter name into the search box to see a set of possible matches.

Do not modify parameters with which you are unfamiliar or have not been instructed to change. Some changes can be harmful to the system. In particular, changing the database connection parameters can break access to the application and the Admin Settings page.

Search:

Tip: You can copy setting names from the documentation to search the available set. Search retrieves matches from the setting name and the help text for the parameter. Do not paste in double quotes from documentation samples.

If your search for a parameter comes up empty and you know that the parameter exists, you must make changes on the Trifacta node in `trifacta-conf.json`. See *Required Platform Configuration*.

Search groupings:

If you search for the following strings, which may appear in property descriptions, you can review groups of settings pertaining to the configuration areas listed below.

NOTE: Do not perform configuration of these areas by simply reviewing and modifying the settings in these parameter groups. Additional configuration may be required. Some required settings may not be grouped, and some of these settings may not be documented. Please review the related documentation sections.

Search string	Setting group
[CORE]	Core platform settings.
[DISTRO]	Settings pertaining to specific distributions. See <i>Configure for Cloudera</i> .
[CLUSTER]	Settings that affect how the platform interacts with the integrated backend cluster. See <i>Prepare Hadoop for Integration with the Platform</i> .
[HIVE]	Settings pertaining to integration with the connected instance of Hive. See <i>Configure for Hive</i> .
[HA]	Settings pertaining to integration with cluster high availability for the Designer Cloud powered by Trifacta platform . See <i>Enable Integration with Cluster High Availability</i> .
[SECURITY]	General settings pertaining to security. See <i>Configure Security</i> .
[SSL]	Settings pertaining to SSL access to the platform. See the SSL section in <i>Configure Security</i> .
[ADVANCED]	Advanced settings.

When you modify a setting, your change is validated against the data type or set of accepted values. String-based entries cannot be validated.

Notes:

- Sensitive information is obfuscated in the display values in the Admin Settings page.
- To save changes, click **Save**.

NOTE: Saving changes forces an automatic type validation of the configuration and a restart of the platform, which terminates any active user sessions.

NOTE: Environmental validation is not performed as part of changes in this user interface. For example, you can change the port number for the Designer Cloud application to an invalid value and save the configuration change. However, when the platform is restarted, the application fails to start, and you cannot continue. In this case, you must fix the problem in `trifacta-conf.json`.

External Service Settings

AWS EMR Cluster ID

If you have deployed your instance of the Designer Cloud powered by Trifacta platform on to Amazon Web Services (AWS) and are connected to an Elastic Map Reduce (EMR) cluster, you can review or modify the cluster identifier in this location. For example, in the event of prolonged outage or failure of the original cluster, you can insert the cluster ID of a secondary cluster to effectively failover to the new cluster.

NOTE: When you first install and integrate with an EMR cluster, this identifier is stored in the Trifacta database for you. It should be modified only if you need to switch to a different EMR cluster. Only one EMR cluster can be active at any time.

NOTE: If this cluster ID is modified, you must modify any other EMR-related settings to match the corresponding values for the new cluster. Please search for `emr` among the admin settings.

When you have entered a new cluster ID, click **Save**.

NOTE: For this setting, you do not have to click the Save button at the bottom of the screen, which restarts the Designer Cloud powered by Trifacta platform .

AWS Region

Enter the AWS region code where the EMR cluster is hosted. For example:

```
us-east-1
```

For a list of available regions, see <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html#concepts-available-regions>.

Resource Bucket

The name of the default S3 bucket where platform resources are stored

Resource Path

The path in the default S3 bucket to where resources are stored

After you have made any changes to the AWS properties, click **Save**.

Services

You can review overall status of the Designer Cloud powered by Trifacta platform .

View Logs

Click the View Logs link to review and download the logs maintained on the Trifacta node.

For more information on these logs, see *System Services and Logs*.

For more information, see *Configure Logging for Services*.

Tricheck

Tricheck performs a variety of tests of your environment to determine its suitability for use with the Designer Cloud powered by Trifacta platform .

Tip: Tricheck should be run immediately after the Trifacta software has been installed or upgraded or whenever there are significant changes to the node or its connected cluster.

Checks include but are not limited to:

- Sufficient hardware resources on the Trifacta node
- Supported versions of software installed on the Trifacta node
- Access to required ports and all nodes of the cluster
- Trifacta node system profiling

NOTE: Tricheck performs no data-dependent checking. It cannot assess suitability of the environment for specific data volumes, connections, or data types.

Click **Run Tricheck** to run checks and download the output log.

SMTP settings

Use this option to send a test email to the specified address through the SMTP email server that has been configured for the Designer Cloud application to use.

NOTE: The SMTP email server to use must be configured. For more information, see *Enable SMTP Email Server Integration*.

Steps:

1. To test the configured settings, click **Check email (SMTP) settings**.
2. Enter a valid email address. Then, click **Check settings**.
3. If the SMTP server is configured properly, a test email is sent to the specified email address.

Upload License

NOTE: For more information on acquiring an updated license file, please contact *Alteryx Support*.

You can update the license file stored on the Trifacta node. Click **Upload License** to browse for and select the license file.

NOTE: By default, the platform checks for a valid license once per hour. To apply your uploaded license immediately, please restart the platform.

For more information on your license, see *License Key*.

Restart

Click **Restart Trifacta** to immediately restart the platform.

Tip: The Restart Trifacta button is the preferred method for restarting the platform. A restart is automatically executed when you save changes to the platform settings.

NOTE: This button may not be available in high availability environments. In those deployments, please restart individual services or use the command line command. For more information, see *Start and Stop the Platform*.

AWS Settings Page

Contents:

- *Workspace Mode*
 - *IAM Role Settings*
 - *AWS Key and Secret Settings*
- *Per-User Mode*
- *Common Settings*
 - *S3 Buckets*
 - *Server-side Encryption*

In the AWS Settings page, workspace administrators can define the AWS credentials mode for the workspace and apply settings for the selected mode, including selecting the credential provider. From the left menu, select **User menu > Admin console > AWS settings**.

NOTE: Before you begin, some information must be gathered from AWS. See *Enable Access to S3 and AWS Resources*.

NOTE: This configuration section applies only if the Designer Cloud powered by Trifacta platform is integrated with Amazon Web Services.

AWS Mode:

Mode	Description
Workspace	<p>In Workspace mode, the workspace administrator applies a single set of AWS credentials for all users in the workspace. These credentials are used by each member of the workspace to authenticate with AWS and to gain access to S3 buckets.</p> <p>Tip: This mode requires more up-front setup but is easy to manage. However, all members of the workspace have the same set of access controls.</p>
Per User	<p>In Per User mode, individual members of the workspace must apply their AWS credentials to their accounts.</p> <p>Tip: This mode is easy to set up but turns responsibility for access controls over to the individual members. If members encounter problems gaining access to S3 assets, the workspace administrator may not be able to troubleshoot them.</p>

Credential Provider:

For workspace or per-user mode, the following provider methods can be used to manage authentication with AWS.

Credential Provider	Description
IAM Role	<p>The Designer Cloud powered by Trifacta platform can use any IAM roles that have been defined for workspace members to access AWS data sources, such as S3 and Redshift.</p> <p>Tip: This credential provider method is recommended.</p>

AWS Key and Secret	You can apply key and secret combinations to gate access to AWS data sources. These combinations can be applied in workspace mode or in per-user mode by individual members.
--------------------	--

Workspace Mode

In workspace mode, you must select the credential provider and then specify the relevant settings.

IAM Role Settings

Prerequisites:

- The IAM roles must include a trust relationship for the Designer Cloud powered by Trifacta platform . For more information, see *Insert Trust Relationship in AWS IAM Role*.
- If you want workspace members to be able to use the on-boarding walkthrough, they must have access to the Trifacta assets required for the walkthrough. For more information, see *Required AWS Account Permissions*.

Apply the following settings to define the IAM roles and related settings.

Setting	Description
Account ID	This value is pre-populated when the workspace is created. <div data-bbox="334 898 1455 978"> NOTE: Do not modify. </div>
External ID	This value is pre-populated when the workspace is created. <div data-bbox="334 1073 1455 1152"> NOTE: Do not modify. </div>
Available IAM Role ARNs	You can specify the set of IAM Role ARNs from which users can select for their access to AWS resources. <div data-bbox="334 1247 1455 1327"> NOTE: These roles cannot be modified if SAML passthrough authentication has been enabled. For more information, see <i>Configure for AWS SAML Passthrough Authentication</i>. </div>
Select Default IAM Role ARN	From the available IAM Role ARNs, you can specify the default one.

AWS Key and Secret Settings

For key-secret authentication to AWS, please specify the following settings.

NOTE: The AWS key and secret must provide read/write access to the default S3 bucket at least.

The account must have the ListAllMyBuckets ACL among its permissions.

Setting	Description
AWS Access Key	The AWS access key to use.
AWS Secret Key	The AWS secret associated with the access key.

Per-User Mode

For per-user mode:

- The workspace administrator must specify the encryption settings only. See below.
- Individual users configure all of the other AWS access settings through the Storage configuration page.

Common Settings

S3 Buckets

For key-secret authentication to AWS, please specify the following settings.

Setting	Description
Default S3 bucket	Specify the name of the default S3 bucket. <div>NOTE: Specify the top-level bucket name only. There should not be any backslashes in your entry.</div>
Extra S3 buckets	You can specify any additional S3 buckets in a comma-separated list of names.

Server-side Encryption

The Designer Cloud powered by Trifacta platform supports the use of server-side encryption when writing results.

NOTE: When encryption is enabled, all buckets to which you are writing must share the same encryption policy.

Setting	Description
Encryption Type	Supported encryption types: <ul style="list-style-type: none">• None• SSE-S3• SSE-KMS
KMS Key ID	If SSE-KMS has been selected, you can paste the KMS Key ID value in this field.

Environment Parameters Page

Through the Environment Parameters page, you can create parameters that are defined for use throughout the workspace or project. Environment parameters can be exported as a set from one environment and imported for use in another.

NOTE: You must be a project owner or workspace administrator to access this page.

Environment parameters			Search parameters
Create ...			
Name	Default value	Last updated	
< > env.bucket_key20210611185636	qa	Today at 9:57 AM	
< > env.year120210611185636	2010	Today at 9:57 AM	
< > env.bucket_key20210611185213	qa	Today at 9:53 AM	
< > env.year120210611185213	2010	Today at 9:53 AM	
< > env.year20210611183222	2010	Today at 9:33 AM	
< > env.bucket_key20210611183222	qa	Today at 9:33 AM	

Figure: Environment Parameters Page

Columns:

- **Name:** Internal name of the environment parameter.
- **Default value:** The default value for the parameter.
- **Last updated:** Timestamp for which the parameter was last modified.

Actions:

- **Create:** Click to create a new environment parameter. Specify:
 - **Name:** Name of the new environment parameter.

NOTE: To distinguish them from other parameters, the prefix `env.` is added to all environment parameter names.

- **Default value:** The default value for the new environment parameter.
- To save your new environment parameter, click **Save**.
- **Import:** Select **Import** to import a ZIP file containing a set of environment parameters that were exported from another environment. For more information, see *Manage Environment Parameters*.
- **Export:** Select **Export** to download a ZIP file containing the definitions and default values for the environment parameters in this environment.
- **Search:** Enter a search string to locate environment parameters by name.


Context menu:

The following options are available for parameters that have been created.

- **Edit value:** Enter a new default value for the environment parameter. Then, click **Save**.
- **Delete:** Delete the environment parameter.

When an environment parameter is deleted, all references to the environment parameter are rendered as an empty string, which may result in broken imported datasets and outputs and other unpredictable issues.

OAuth 2.0 Clients Page

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

Through the OAuth 2.0 Clients page, workspace administrators can create and manage OAuth 2.0 clients for authentication with third-party systems. In the Admin console, select **OAuth 2.0 Clients**.

OAuth 2.0 clients

Register OAuth 2.0 client

Connect to other applications using the OAuth 2.0 authorization framework. [Learn more about OAuth 2.0 client creation.](#)

Name	Type	Last updated
Test2	snowflake	Today at 2:35 PM
Test	snowflake	Today at 2:35 PM

Figure: OAuth 2.0 Clients page

Columns:

- **Name:** Display name of the client.
- **Type:** Type of client. For more information on supported clients, see *Create OAuth2 Client*.
- **Last update:** Timestamp for last modification to the listed client.

Actions:

- To create an OAuth 2.0 client, click **Register OAuth 2.0 Client**. For more information, see *Create OAuth2 Client*.
- To delete a client, select **Delete OAuth 2.0 Client** from the context menu.

Deleting an OAuth 2.0 client cannot be undone. When a client is deleted, any connections that utilized the client no longer work. Datasets and output locations may no longer be accessible through the application.

Workspace Admin Permissions

Contents:

- *Configuration*
 - *User Management*
 - *Object Access*
 - *Data Access*
-

The workspace admin user is super-user for the entire workspace.

NOTE: In Designer Cloud Powered by Trifacta Enterprise Edition, any user who is granted the admin role is also granted the workspace admin role, which enables owner-level access to some object types in the workspace. Details are below.

Configuration

The workspace admin can enable and disable features and capabilities in the workspace. For more information, see *Workspace Settings Page*.

User Management

A workspace admin can administer all of the other users of the workspace, including disabling or deleting the user.

NOTE: In Designer Cloud Powered by Trifacta Enterprise Edition, the workspace admin can also edit the platform roles assigned to individual users.

See *Users Page*.

Object Access

A workspace admin has owner-level access to objects in the workspace.

NOTE: A workspace admin can access these objects like their owners, even if the objects have not been shared.

This access applies, but is not limited, to the following types of objects:

- Flows
- Connections (see below)
- Output objects
- Job profiles and results
- Plans and tasks

A workspace admin has collaborator-level access to the following objects:

- Imported datasets
- Macros
- Schedules

A workspace admin does not have any changed permissions for access to the following object types:

- Deployments and releases

Data Access

The workspace admin can access the data of individual users under the following conditions.

NOTE: Workspace admin privileges do not affect access permissions on outside storage systems. Those systems can prevent data access by the workspace admin user.

Connections with credentials:

If the data is accessed through a connection that requires a specific set of credentials, then the workspace admin can access all data available through the connection when the credentials are shared.

If connection credential sharing is disabled after a connection has already been shared with credentials, then the connection remains accessible to the workspace admin and to all users who were previously shared the connection. Workspace admins created in the future also inherit this access. The sharing of a connection's credentials cannot be revoked, except by deleting the connection.

A workspace admin cannot:

1. Modify or remove the shared credentials.
2. Change the credential sharing on another user's connection.

If a connection with shared credentials remains after credential sharing has been disabled, you can do one of the following for the connection:

- Edit the connection to use credentials that are safe to share with all affected users.
- Create a duplicate connection with private credentials. Delete the old shared connection.

For more information on credential sharing, see *Configure Sharing*.

File-based backend storage:
Source datasets and job results that are stored on file-based backend storage systems for individual users can be accessed by the workspace admin except in the following situations:

- If users have user-specific access controls to the storage, such as secure impersonation, the workspace admin can only access a user's data if the admin's own permissions enable it.
- Directory permissions on user directories may prevent the workspace admin from accessing a user's data. For example, the workspace admin user can see the link to a user's job results that were written on the backend storage. However, when the workspace admin attempts to download those results, a permissions error is displayed, since the workspace admin user does not have permissions on the directory.

Relational connections:

Data is accessible under the connections with credentials limitations described above.

Admin Download Logs Dialog

Contents:

- *Logs by Timeframe*
- *Logs by Job ID*
- *Logs by Session ID*
- *Download*

Administrators of Designer Cloud Powered by Trifacta® Enterprise Edition can download log files based on a user's session identifier, a job identifier, or across the Designer Cloud powered by Trifacta platform for a specified time period. From the Resources menu, select **Download logs**. The data downloaded from this dialog is encrypted by default.

NOTE: The files download through this dialog are always unencrypted.

NOTE: For more information on disabling this feature, see *Configure Support Bundling*.

Non-administrators can download logs for their current session. For more information, see *Download Logs Dialog*.

Download logs

Collect logs by

Time frame

Collect all log files available for a specific time period

Time frame

Last

2

Hours

Log file size limit ?

1000000

Bytes

Cancel

Download logs

Figure: Download Logs Dialog for Admins

Collect logs by: Select the method by which log files for the Designer Cloud powered by Trifacta platform are collected.

Log file size limit: You can specify the size limit of individual log files in bytes. The default size is 1 MB.

Logs by Timeframe

You can download logs by specific time period. You can select the last few hours or even customize the date range to download the log files.

Tip: Try to narrow the time frame if possible. Larger time frames are more likely to run up against the size limit for individual log files.

Steps:

1. To download the log files by time frame, select the Time frame option from the **Collect logs by** drop-down.
2. Select the required option from the **Time frame** drop-down. The following are the available options:
 - a. **Last:** Preceding number of minutes, hours, or days.
 - b. **Between:** Use the date and time tools to specify the starting (top) and ending (bottom) dates for the range.
3. To control the size of the log files, you can specify the size limit of individual log files in bytes. The default size is 1 MB.
4. To download logs, click **Download logs**. Logs are downloaded as a ZIP file.

Logs by Job ID

You can download log files for a specific job ID.

Tip: Administrators can review the IDs for all accessible jobs in the Job History page. See *Job History Page*.

Steps:

1. To download the log files by Job ID, select the Job ID option from the **Collect logs by** drop-down.
2. In the Job ID field, enter the required job ID.

NOTE: The Job ID is a string of numbers that can be found in the Job History page or Job emails.

3. To control the size of the log files, you can specify the size limit of individual log files in bytes. The default size is 1 MB.
4. To download logs, click **Download logs**. Logs are downloaded as a ZIP file.

Logs by Session ID

You can download the logs files for a specified user session.

Tip: Non-admin users can retrieve their session ID from the application. For more information, see *Download Logs Dialog*.

Steps:

1. To download the log files by Job ID, select the Job ID option from the **Collect logs by** drop-down.
2. From the Session ID drop-down, select any one of the available options:
 - a. **I've got a session ID:** If you know the session ID, enter the same in the **Enter Session ID** field.
 - b. **Use my current session ID:** If you want to use your current session ID, select this option and the current session ID is automatically populated.
3. To control the size of the log files, you can specify the size limit of individual log files in bytes. The default size is 1 MB.
4. To download logs, click **Download logs**. Logs are downloaded as a ZIP file.

Download

To download the specified set of logs, click **Download logs**.

For more information on the contents of this download, see *Support Bundle Contents*.

For more information on configuring the contents of the support bundle, see *Configure Support Bundling*.

Required AWS Account Permissions

Contents:

- S3
 - *Read-only access policies*
 - *Write access policies*
 - *Other AWS policies for S3*
- Redshift
- Snowflake
 - *Staging bucket*
 - *Snowflake bucket*
 - *Snowflake running environment*
- EMR

To access the following AWS resources, you must configure your AWS account or accounts with the listed permissions. These permissions can be applied through AWS access key/secret combinations or through IAM roles applied to the account.

S3

All access to S3 sources occurs through a single AWS account (system mode) or through an individual user's account (user mode). For either mode, the AWS access key and secret combination must provide access to the default bucket associated with the account.

NOTE: These permissions should be set up by your AWS administrator.

Read-only access policies

NOTE: To enable viewing and browsing of all folders within a bucket, the following permissions are required:

- The system account or individual user accounts must have the `ListAllMyBuckets` access permission for the bucket.
- All objects to be browsed within the bucket must have Get access enabled.

The policy statement to enable read-only access to your default S3 bucket should look similar to the following. Replace `3c-my-s3-bucket` with the name of your bucket:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Resource": [
        "arn:aws:s3:::3c-my-s3-bucket",
        "arn:aws:s3:::3c-my-s3-bucket/*",
      ]
    }
  ]
}
```

```

    }
  ]
}

```

Write access policies

Write access is enabled by adding the `PutObject` and `DeleteObject` actions to the above. Replace `3c-my-s3-bucket` with the name of your bucket:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket",
        "s3:GetBucketLocation",
        "s3:PutObject",
        "s3>DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::3c-my-s3-bucket",
        "arn:aws:s3:::3c-my-s3-bucket/*"
      ]
    }
  ]
}

```

Other AWS policies for S3

Policy for access to Trifacta public buckets

To access S3 assets that are created by Alteryx®, you must apply the following policy definition to any IAM role that is used to access the Designer Cloud powered by Trifacta platform . These buckets contain demo assets:

NOTE: Product walkthroughs must be enabled. For more information, see *Workspace Settings Page*.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::trifacta-public-datasets/*",
        "arn:aws:s3:::trifacta-public-datasets"
      ]
    }
  ]
}

```

For more information on creating policies, see <https://console.aws.amazon.com/iam/home#/policies>.

KMS policy

If any accessible bucket is encrypted with KMS-SSE, another policy must be deployed. For more information, see <https://docs.aws.amazon.com/kms/latest/developerguide/iam-policies.html>.

Attribute-based access to S3

If you are using IAM roles to provide per-user access to S3, you can apply AWS session tags to any request for S3 resources, which allows you to leverage your enterprise permissioning to S3 based on the user identifier. IAM policies must be modified, and this feature must be enabled. For more information, see *Configure AWS Per-User Auth for Temporary Credentials*.

Redshift

Since Redshift requires S3 to be used, to enable read/write access to Redshift using an IAM role, the sole additional requirement to the above is to add the `GetClusterCredentials` permission to the IAM role used for S3. A policy statement similar to the following example needs to be included as part of any IAM role used by the Designer Cloud powered by Trifacta platform users to access AWS resources.

The following example policy adds the `GetClusterCredentials` permission for the specified AWS user (`aws:userid`). This user is permitted to get cluster credentials for three different resources:

- a personal Redshift cluster
- The `testdb` cluster
- The `common_group` cluster

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GetClusterCredsStatement",
      "Effect": "Allow",
      "Action": [
        "redshift:GetClusterCredentials"
      ],
      "Resource": [
        "arn:aws:redshift:us-west-2:123456789012:dbuser:examplecluster/${redshift:DbUser}",
        "arn:aws:redshift:us-west-2:123456789012:dbname:examplecluster/testdb",
        "arn:aws:redshift:us-west-2:123456789012:dbgroup:examplecluster/common_group"
      ],
      "Condition": {
        "StringEquals": {
          "aws:userid": "AIDIODR4TAW7CSEXAMPLE:${redshift:DbUser}@yourdomain.com"
        }
      }
    }
  ]
}
```

For more information on `getClusterCredentials`, see https://docs.aws.amazon.com/redshift/latest/APIReference/API_GetClusterCredentials.html.

Snowflake

If you are creating a connection to your AWS-based Snowflake deployment, you must specify the following policies in the operative IAM role(s) for each S3 bucket:

Staging bucket

If you are creating your own Snowflake stage, it must point to the default S3 bucket in use by the Designer Cloud powered by Trifacta platform . The policy that you created for read-write access to S3 should be applied to the Snowflake user.

NOTE: If users in your deployment are using IAM roles in user mode for AWS access, then the Snowflake stage must have permissions to write to the user's S3 bucket.

Snowflake bucket

You must create a separate policy to permit access to the S3 bucket that backs your AWS-based Snowflake deployment. The following example permission provides the minimum set of permissions.

Notes:

- The `s3:GetBucketLocation` is required for access to the S3 bucket that Snowflake requires for itself.
- The additional `s3:PutObject` and `s3>DeleteObject` permissions are required only if you plan to unload files to the bucket or automatically purge the files after loading them into a table.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3>DeleteObject",
        "s3>DeleteObjectVersion",
        "s3:GetBucketLocation"
      ],
      "Resource": "arn:aws:s3:::<snowflake_bucket_name>/<prefix>/*"
    },
    {
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::<snowflake_bucket_name>",
      "Condition": {
        "StringLike": {
          "s3:prefix": [
            "<prefix>/*"
          ]
        }
      }
    }
  ]
}
```

Where:

- `<snowflake_bucket_name>` = the name of the S3 bucket that is used by Snowflake
- `<prefix>` = the folder path prefix within the bucket. This value can be omitted if it is not required.
 - The above `StringLike` definition grants access to all prefixes on the bucket.

NOTE: If your bucket or prefixed path contains more than 1000 files, you may encounter the following error: Access Denied (Status Code: 403; Error Code: AccessDenied) .

- To address the above error, specify the `StringLike` condition with the following change. This change allows access to all files while eliminating the condition that causes the above error:

```
"Condition": {
  "StringLike": {
    "s3:prefix": [
      "*"
    ]
  }
}
```

For more information, see <https://docs.snowflake.com/en/user-guide/data-load-s3-config-aws-iam-user.html>.

Snowflake running environment

If the above policy has been deployed for the S3 buckets used for Snowflake and publishing is enabled, then no further permissions are required.

NOTE: Additional limitations may apply.


EMR

Additional permissions to access EMR depend on how the Trifacta deployment is configured to interact with EMR. For more information, see *Configure for EMR*.

Privileges and Roles Reference

Contents:

- Privileges
 - Flows
 - Connections
 - Plans
- Standard Roles
 - default
 - Workspace admin

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

In the Designer Cloud application , you can create and assign roles, each of which consists of one or more privileges. A **privilege** is a level of access to a type of object.

Below, you can review the available privileges, including the supported levels for each.

For more information on privileges and roles, see *Overview of Authorization*.

Privileges


Flows

The flows privilege governs access to flow objects.

Access Level	Name	Description
0	none	Assigned role cannot see or use flows, including the pages where flows are available.
1	viewer	Assigned user can access Flows page and Flow View page for flows that the user owns or has been shared. User can also run jobs on the user's own flows. User cannot make changes to any flows.
2	editor	All of the above, plus: Assigned user can edit, share, and run jobs on flows to which the user has access. <div>NOTE: By default, editors can also schedule flows. This option can be disabled by an administrator.</div> <div>Tip: Flow editors can edit any custom SQL used to import datasets into the flow.</div>
3	author	All of the above, plus: Assigned user can create new flows, schedule flows, and delete flows.

Tip: If you have enabled deployment management, a deployment user should be assigned author-level access. Lesser flow roles may prevent the deployment user from properly importing and managing flows. See *Roles Page*.


Connections

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

The connections privilege governs access to connection objects.

Access Level	Name	Description
0	none	Assigned role cannot see or use connections, including the pages where connections are available.
1	viewer	Assigned user can access Connections page for connections that the user owns or has been shared. User can share connections. User cannot make changes to any connections.
2	editor	All of the above, plus: Assigned user can edit and share connections to which the user has access.
3	author	All of the above, plus: Assigned user can create new connections and delete connections.

Plans

 **Feature Availability:** This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

The plans privilege manages access to plan objects.

Access Level	Name	Description
0	none	Assigned role cannot see or use plans, including the pages where plans are available.
1	viewer	Assigned user can access Plans page and Plan View page for plans that the user owns or has been shared. User can also run jobs on the user's own plans. User can cancel plan runs.
2	editor	All of the above, plus: Assigned user can edit, share, and run jobs on plans to which the user has access. <div>NOTE: By default, editors can also schedule plans. This option can be disabled by an administrator.</div>
3	author	All of the above, plus: Assigned user can create new plans, schedule plans, and delete plans.

Standard Roles

The following roles are provided with the product.

NOTE: The following roles cannot be removed.

default

The default role is assigned to each user when the user is initially created. This role contains the following permissions:

Privilege	Access Level/Name
Flows	3 - author
Connections	3 - author
Plans	3 - author
User defined functions	3 - author

Tip: You can modify the default role if you want to set a lower level of base access for each new user of the product. For more information, see *Overview of Authorization*.

Workspace admin

This role provides super-user privileges to the assigned user.

NOTE: This role enables for the user owner-level access to all objects in the project or workspace and access to all admin-level settings and configuration pages in the admin console. This role should not be assigned to many users. At least one user should always have this role.

NOTE: You cannot modify or delete this role.

Platform Reference

The following reference information pertains to the Designer Cloud powered by Trifacta® platform .

Supported SQL Syntax

Contents:

- *Basic Syntax*
 - *Supported syntax by datastore*
 - *General Examples*
 - *Column aliasing*
 - *Collect whole table*
 - *Filter columns*
 - *Filter rows*
 - *Multi-line statement for imported datasets*
-



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

This section provides general information on how the Designer Cloud Powered by Trifacta Enterprise Edition uses SQL to interact with your databases, including syntax requirements and examples.

Basic Syntax

Your SQL statements must be valid for the syntax expected by the target relational system. In particular, object delimiters may vary between systems.

NOTE: The proper syntax depends on your database system. Please consult the documentation for your product for details.

Tip: Although some relational systems do not require object delimiters around column names, it is recommended that you add them to all applicable objects.

Tip: Avoid using column type identifiers (e.g. `int`) and other SQL keywords as object names. Some systems may generate invalid SQL errors.

NOTE: In the following sections, Oracle syntax is used in the examples. Please modify the examples for your target system.

Supported syntax by datastore

Individual datastores may have differences in the supported syntax. For more information, please see the documentation for your datastore.

General Examples

Here are some basic SQL examples to get started.

Column aliasing

If your select statement results in multiple columns with same name, the query fails to validate or fails on execution, such as selecting all columns in a JOIN. In these cases, columns must be properly aliased.

NOTE: This error will be caught either during validating or during dataset import.

For example, in the following JOIN, the EMPLOYEE and DEPARTMENT tables have column names department_id and department_id.

```
SELECT * FROM EMPLOYEE INNER JOIN DEPARTMENT ON (department_id = department_id);
```

The above query generates an error. Columns must be properly aliased, as in the following:

```
SELECT e.id, e.department_id, e.first_name, e.last_name, d.department_name FROM EMPLOYEE AS E INNER JOIN  
DEPARTMENT d ON (e.department_id = d.department_id);
```

Collect whole table

```
SELECT * FROM "DB1"."table2";
```

Filter columns

```
SELECT lastName,firstName FROM "DB1"."table2";
```

Filter rows

```
SELECT lastName,firstName FROM "DB1"."table2" WHERE invoiceAmt > 10000;
```

Multi-line statement for imported datasets



Feature Availability: This feature may not be available in all product editions. For more information on available features, see *Compare Editions*.

The following example uses a multi-line SQL sequence to import a dataset:

NOTE: Multi-line SQL support is considered an advanced use case. This feature must be enabled.

The following example inserts values in the TABLE_INVENTORY table and then queries the table. It utilizes Oracle syntax:

```
INSERT INTO "SALES"."TABLE_INVENTORY" ("ID", "AVAILABILITY") VALUES (1, 10);  
SELECT * FROM "SALES"."TABLE_INVENTORY";
```

Locale Settings

The Designer Cloud powered by Trifacta® platform supports a tier-based scheme for applying locale settings.

NOTE: Locale settings apply only to the inference and validation of data in the Designer Cloud application . Underlying data is not affected by changing locale.

NOTE: After saving changes to your locale, refresh your page. Subsequent executions of the data inference service use the new locale settings.

NOTE: When locale is changed, data type validation is affected only on subsequent executions of the data type inference service. If you are using structured datasets, such as schematized JDBC sources, data types may be attached to the datasets that you have already imported. These data types are not affected.

Locale settings can be configured for:

- **Project or workspace level:** global local for all users in the environment
- **User level:** In User Preferences, individual users can set their personal locale.

Supported Locales

The following locales are available:

- United States
- Australia
- Austria
- Belgium
- Bulgaria
- Canada
- Croatia
- Cyprus
- Czech Republic
- Denmark
- Estonia
- Finland
- France
- Germany
- Hungary
- Greece
- Ireland
- Italy
- Japan
- Latvia
- Lithuania
- Luxembourg
- Malta
- New Zealand
- Netherlands
- Poland
- Portugal
- Romania

- Singapore
- Slovakia
- Slovenia
- Spain
- Sweden
- United Kingdom

Product Areas Affected by Locale

Datetime

- Data type inference more accurately recognizes Datetime values based on locale settings.
- Suggestion cards containing Datetime transformations and example previews utilize locale settings.
- Number of weeks in the year may vary. For example, the WEEKNUM function calculated on the last few dates of the calendar year may return different values depending on the locale where the calculation is performed:

NOTE: The output calculations for number of weeks may vary between browser, Trifacta Photon, and the running environment where the job may be executed, based upon locale.

- **EN-US:** Maximum of 52 weeks
- **ISO 8601:** Maximum of 53 weeks
- You can add steps similar to the following, which adds a 53rd week as a possible output for WEEKNUM:

```
derive type: single value: dateformat(date(year(myDate), 1, 1), &apos;yyyy\MM\dd&apos;) as: &apos;NewYearsDayforMyYear&apos;
```

```
derive type: single value: if(datedif(NewYearsDayforMyYear, myDate, day) > (52 * 7), 53, weeknum(myDate)) as: &apos;weekNumforMyDate&apos;
```

Supported Time Zone Values

This section provides information about supported time zone values in the Designer Cloud powered by Trifacta® platform

American Time Zones

The following American time zones are mapped to the time zone values supported in the Designer Cloud powered by Trifacta platform :

Tip: The **Designer Cloud powered by Trifacta platform** does make adjustments for Daylight Savings Time.

US Time Zone	Time Zone Value	Standard Time (UTC)	Daylight Savings Time (UTC)	Included States and Territories	Notes
Atlantic	America / Puerto Rico	UTC-04:00	n/a	Puerto Rico, US Virgin Islands	Daylight Savings Time is not observed.
Eastern	US / Eastern	UTC-05:00	UTC-04:00	Entire: Connecticut, Delaware, Georgia, Maine, Maryland, Massachusetts, New Hampshire, New Jersey, New York, North Carolina, Ohio, Pennsylvania, Rhode Island, South Carolina, Vermont, Virginia, West Virginia Partial: Florida, Indiana, Kentucky, Michigan, Tennessee	
Central	US / Central	UTC-06:00	UTC-05:00	Entire: Alabama, Arkansas, Illinois, Iowa, Louisiana, Minnesota, Mississippi, Missouri, Oklahoma, Wisconsin Partial: Florida, Indiana, Kansas, Kentucky, Michigan, Nebraska, North Dakota, South Dakota, Tennessee, Texas	
Mountain	US / Mountain	UTC-07:00	UTC-06:00	Entire: Arizona, Colorado, Montana, New Mexico, Utah, Wyoming Partial: Idaho, Kansas, Nebraska, Nevada, North Dakota, Oregon, South Dakota, Texas	Most of Arizona does not observe Daylight Savings Time. Use US / Arizona in this area.
Pacific	US / Pacific	UTC-08:00	UTC-07:00	Entire: California, Washington Partial: Idaho, Nevada, Oregon	
Alaska	US / Alaska	UTC-09:00	UTC-08:00	Partial: Alaska	Daylight Savings Time is not observed in the Aleutian Islands. Use US / Aleutian in this area.
Hawaii	US / Hawaii	UTC-10:00	UTC-09:00	Entire: Hawaii Partial: Alaska	Daylight Savings Time is not observed in Hawaii.

Global Time Zone Values

For the functions that support use of specified time zones, you can apply the following string values as parameters to specify the time zone:

Time Zone Value	Standard time (UTC)	Daylight Saving time (UTC)	Partial country coverage
Africa/Abidjan	UTC+00:00	UTC+00:00	
Africa/Accra	UTC+00:00	UTC+00:00	
Africa/Addis_Ababa	UTC+03:00	UTC+03:00	
Africa/Algiers	UTC+01:00	UTC+01:00	
Africa/Asmara	UTC+03:00	UTC+03:00	
Africa/Bamako	UTC+00:00	UTC+00:00	
Africa/Bangui	UTC+01:00	UTC+01:00	
Africa/Banjul	UTC+00:00	UTC+00:00	
Africa/Bissau	UTC+00:00	UTC+00:00	
Africa/Blantyre	UTC+02:00	UTC+02:00	
Africa/Brazzaville	UTC+01:00	UTC+01:00	
Africa/Bujumbura	UTC+02:00	UTC+02:00	
Africa/Cairo	UTC+02:00	UTC+02:00	
Africa/Casablanca	UTC+01:00	UTC+01:00	
Africa/Ceuta	UTC+01:00	UTC+01:00	Ceuta, Melilla
Africa/Conakry	UTC+00:00	UTC+00:00	
Africa/Dakar	UTC+00:00	UTC+00:00	
Africa/Dar_es_Salaam	UTC+03:00	UTC+03:00	
Africa/Djibouti	UTC+03:00	UTC+03:00	
Africa/Douala	UTC+01:00	UTC+01:00	
Africa/El_Aaiun	UTC+00:00	UTC+01:00	
Africa/Freetown	UTC+00:00	UTC+00:00	
Africa/Gaborone	UTC+02:00	UTC+02:00	
Africa/Harare	UTC+02:00	UTC+02:00	
Africa/Johannesburg	UTC+02:00	UTC+02:00	
Africa/Juba	UTC+03:00	UTC+03:00	
Africa/Kampala	UTC+03:00	UTC+03:00	
Africa/Khartoum	UTC+02:00	UTC+02:00	
Africa/Kigali	UTC+02:00	UTC+02:00	
Africa/Kinshasa	UTC+01:00	UTC+01:00	Dem. Rep. of Congo (west)
Africa/Lagos	UTC+01:00	UTC+01:00	
Africa/Libreville	UTC+01:00	UTC+01:00	
Africa/Lome	UTC+00:00	UTC+00:00	
Africa/Luanda	UTC+01:00	UTC+01:00	

Africa/Lubumbashi	UTC+02:00	UTC+02:00	Dem. Rep. of Congo (east)
Africa/Lusaka	UTC+02:00	UTC+02:00	
Africa/Malabo	UTC+01:00	UTC+01:00	
Africa/Maputo	UTC+02:00	UTC+02:00	
Africa/Maseru	UTC+02:00	UTC+02:00	
Africa/Mbabane	UTC+02:00	UTC+02:00	
Africa/Mogadishu	UTC+03:00	UTC+03:00	
Africa/Monrovia	UTC+00:00	UTC+00:00	
Africa/Nairobi	UTC+03:00	UTC+03:00	
Africa/Ndjamena	UTC+01:00	UTC+01:00	
Africa/Niamey	UTC+01:00	UTC+01:00	
Africa/Nouakchott	UTC+00:00	UTC+00:00	
Africa/Ouagadougou	UTC+00:00	UTC+00:00	
Africa/Porto-Novo	UTC+01:00	UTC+01:00	
Africa/Sao_Tome	UTC+00:00	UTC+00:00	
Africa/Timbuktu	UTC+00:00	UTC+00:00	
Africa/Tripoli	UTC+02:00	UTC+02:00	
Africa/Tunis	UTC+01:00	UTC+01:00	
Africa/Windhoek	UTC+02:00	UTC+02:00	
America/Adak	UTC10:00	UTC09:00	Aleutian Islands
America/Anchorage	UTC09:00	UTC08:00	Alaska (most areas)
America/Anguilla	UTC04:00	UTC04:00	
America/Antigua	UTC04:00	UTC04:00	
America/Araguaina	UTC03:00	UTC03:00	Tocantins
America/Argentina /Buenos_Aires	UTC03:00	UTC03:00	Buenos Aires (BA, CF)
America/Argentina /Catamarca	UTC03:00	UTC03:00	Catamarca (CT); Chubut (CH)
America/Argentina /ComodRivadavia	UTC03:00	UTC03:00	
America/Argentina/Cordoba	UTC03:00	UTC03:00	Argentina (most areas: CB, CC, CN, ER, FM, MN, SE, SF)
America/Argentina/Jujuy	UTC03:00	UTC03:00	Jujuy (JY)
America/Argentina/La_Rioja	UTC03:00	UTC03:00	La Rioja (LR)
America/Argentina/Mendoza	UTC03:00	UTC03:00	Mendoza (MZ)
America/Argentina /Rio_Gallegos	UTC03:00	UTC03:00	Santa Cruz (SC)
America/Argentina/Salta	UTC03:00	UTC03:00	Salta (SA, LP, NQ, RN)
America/Argentina /San_Juan	UTC03:00	UTC03:00	San Juan (SJ)
America/Argentina/San_Luis	UTC03:00	UTC03:00	San Luis (SL)
America/Argentina/Tucuman	UTC03:00	UTC03:00	Tucuman (TM)
America/Argentina/Ushuaia	UTC03:00	UTC03:00	Tierra del Fuego (TF)
America/Aruba	UTC04:00	UTC04:00	

America/Asuncion	UTC04:00	UTC03:00	
America/Atikokan	UTC05:00	UTC05:00	EST - ON (Atikokan); NU (Coral H)
America/Atka	UTC10:00	UTC09:00	
America/Bahia	UTC03:00	UTC03:00	Bahia
America/Bahia_Banderas	UTC06:00	UTC05:00	Central Time - Bahia de Banderas
America/Barbados	UTC04:00	UTC04:00	
America/Belem	UTC03:00	UTC03:00	Para (east); Amapa
America/Belize	UTC06:00	UTC06:00	
America/Blanc-Sablon	UTC04:00	UTC04:00	AST - QC (Lower North Shore)
America/Boa_Vista	UTC04:00	UTC04:00	Roraima
America/Bogota	UTC05:00	UTC05:00	
America/Boise	UTC07:00	UTC06:00	Mountain - ID (south); OR (east)
America/Buenos_Aires	UTC03:00	UTC03:00	
America/Cambridge_Bay	UTC07:00	UTC06:00	Mountain - NU (west)
America/Campo_Grande	UTC04:00	UTC03:00	Mato Grosso do Sul
America/Cancun	UTC05:00	UTC05:00	Eastern Standard Time - Quintana Roo
America/Caracas	UTC04:00	UTC04:00	
America/Catamarca	UTC03:00	UTC03:00	
America/Cayenne	UTC03:00	UTC03:00	
America/Cayman	UTC05:00	UTC05:00	
America/Chicago	UTC06:00	UTC05:00	Central (most areas)
America/Chihuahua	UTC07:00	UTC06:00	Mountain Time - Chihuahua (most areas)
America/Coral_Harbour	UTC05:00	UTC05:00	
America/Cordoba	UTC03:00	UTC03:00	
America/Costa_Rica	UTC06:00	UTC06:00	
America/Creston	UTC07:00	UTC07:00	MST - BC (Creston)
America/Cuiaba	UTC04:00	UTC03:00	Mato Grosso
America/Curacao	UTC04:00	UTC04:00	
America/Danmarkshavn	UTC+00:00	UTC+00:00	National Park (east coast)
America/Dawson	UTC08:00	UTC07:00	Pacific - Yukon (north)
America/Dawson_Creek	UTC07:00	UTC07:00	MST - BC (Dawson Cr, Ft St John)
America/Denver	UTC07:00	UTC06:00	Mountain (most areas)
America/Detroit	UTC05:00	UTC04:00	Eastern - MI (most areas)
America/Dominica	UTC04:00	UTC04:00	
America/Edmonton	UTC07:00	UTC06:00	Mountain - AB; BC (E); SK (W)
America/Eirunepe	UTC05:00	UTC05:00	Amazonas (west)
America/El_Salvador	UTC06:00	UTC06:00	
America/Ensenada	UTC08:00	UTC07:00	
America/Fort_Nelson	UTC07:00	UTC07:00	MST - BC (Ft Nelson)
America/Fort_Wayne	UTC05:00	UTC04:00	

America/Fortaleza	UTC03:00	UTC03:00	Brazil (northeast: MA, PI, CE, RN, PB)
America/Glace_Bay	UTC04:00	UTC03:00	Atlantic - NS (Cape Breton)
America/Godthab	UTC03:00	UTC02:00	Greenland (most areas)
America/Goose_Bay	UTC04:00	UTC03:00	Atlantic - Labrador (most areas)
America/Grand_Turk	UTC05:00	UTC04:00	
America/Grenada	UTC04:00	UTC04:00	
America/Guadeloupe	UTC04:00	UTC04:00	
America/Guatemala	UTC06:00	UTC06:00	
America/Guayaquil	UTC05:00	UTC05:00	Ecuador (mainland)
America/Guyana	UTC04:00	UTC04:00	
America/Halifax	UTC04:00	UTC03:00	Atlantic - NS (most areas); PE
America/Havana	UTC05:00	UTC04:00	
America/Hermosillo	UTC07:00	UTC07:00	Mountain Standard Time - Sonora
America/Indiana/Indianapolis	UTC05:00	UTC04:00	Eastern - IN (most areas)
America/Indiana/Knox	UTC06:00	UTC05:00	Central - IN (Starke)
America/Indiana/Marengo	UTC05:00	UTC04:00	Eastern - IN (Crawford)
America/Indiana/Petersburg	UTC05:00	UTC04:00	Eastern - IN (Pike)
America/Indiana/Tell_City	UTC06:00	UTC05:00	Central - IN (Perry)
America/Indiana/Vevay	UTC05:00	UTC04:00	Eastern - IN (Switzerland)
America/Indiana/Vincennes	UTC05:00	UTC04:00	Eastern - IN (Da, Du, K, Mn)
America/Indiana/Winamac	UTC05:00	UTC04:00	Eastern - IN (Pulaski)
America/Indianapolis	UTC05:00	UTC04:00	
America/Inuvik	UTC07:00	UTC06:00	Mountain - NT (west)
America/Iqaluit	UTC05:00	UTC04:00	Eastern - NU (most east areas)
America/Jamaica	UTC05:00	UTC05:00	
America/Jujuy	UTC03:00	UTC03:00	
America/Juneau	UTC09:00	UTC08:00	Alaska - Juneau area
America/Kentucky/Louisville	UTC05:00	UTC04:00	Eastern - KY (Louisville area)
America/Kentucky/Monticello	UTC05:00	UTC04:00	Eastern - KY (Wayne)
America/Knox_IN	UTC06:00	UTC05:00	
America/Kralendijk	UTC04:00	UTC04:00	
America/La_Paz	UTC04:00	UTC04:00	
America/Lima	UTC05:00	UTC05:00	
America/Los_Angeles	UTC08:00	UTC07:00	Pacific
America/Louisville	UTC05:00	UTC04:00	
America/Lower_Princes	UTC04:00	UTC04:00	
America/Maceio	UTC03:00	UTC03:00	Alagoas, Sergipe
America/Managua	UTC06:00	UTC06:00	
America/Manaus	UTC04:00	UTC04:00	Amazonas (east)
America/Marigot	UTC04:00	UTC04:00	

America/Martinique	UTC04:00	UTC04:00	
America/Matamoros	UTC06:00	UTC05:00	Central Time US - Coahuila, Nuevo Leon, Tamaulipas (US border)
America/Mazatlan	UTC07:00	UTC06:00	Mountain Time - Baja California Sur, Nayarit, Sinaloa
America/Mendoza	UTC03:00	UTC03:00	
America/Menominee	UTC06:00	UTC05:00	Central - MI (Wisconsin border)
America/Merida	UTC06:00	UTC05:00	Central Time - Campeche, Yucatan
America/Metlakatla	UTC09:00	UTC08:00	Alaska - Annette Island
America/Mexico_City	UTC06:00	UTC05:00	Central Time
America/Miquelon	UTC03:00	UTC02:00	
America/Moncton	UTC04:00	UTC03:00	Atlantic - New Brunswick
America/Monterrey	UTC06:00	UTC05:00	Central Time - Durango; Coahuila, Nuevo Leon, Tamaulipas (most areas)
America/Montevideo	UTC03:00	UTC03:00	
America/Montreal	UTC05:00	UTC04:00	
America/Montserrat	UTC04:00	UTC04:00	
America/Nassau	UTC05:00	UTC04:00	
America/New_York	UTC05:00	UTC04:00	Eastern (most areas)
America/Nipigon	UTC05:00	UTC04:00	Eastern - ON, QC (no DST 196773)
America/Nome	UTC09:00	UTC08:00	Alaska (west)
America/Noronha	UTC02:00	UTC02:00	Atlantic islands
America/North_Dakota /Beulah	UTC06:00	UTC05:00	Central - ND (Mercer)
America/North_Dakota /Center	UTC06:00	UTC05:00	Central - ND (Oliver)
America/North_Dakota /New_Salem	UTC06:00	UTC05:00	Central - ND (Morton rural)
America/Ojinaga	UTC07:00	UTC06:00	Mountain Time US - Chihuahua (US border)
America/Panama	UTC05:00	UTC05:00	
America/Pangnirtung	UTC05:00	UTC04:00	Eastern - NU (Pangnirtung)
America/Paramaribo	UTC03:00	UTC03:00	
America/Phoenix	UTC07:00	UTC07:00	MST - Arizona (except Navajo)
America/Port_of_Spain	UTC04:00	UTC04:00	
America/Port-au-Prince	UTC05:00	UTC04:00	
America/Porto_Acre	UTC05:00	UTC05:00	
America/Porto_Velho	UTC04:00	UTC04:00	Rondonia
America/Puerto_Rico	UTC04:00	UTC04:00	
America/Punta_Arenas	UTC03:00	UTC03:00	Region of Magallanes
America/Rainy_River	UTC06:00	UTC05:00	Central - ON (Rainy R, Ft Frances)
America/Rankin_Inlet	UTC06:00	UTC05:00	Central - NU (central)
America/Recife	UTC03:00	UTC03:00	Pernambuco
America/Regina	UTC06:00	UTC06:00	CST - SK (most areas)
America/Resolute	UTC06:00	UTC05:00	Central - NU (Resolute)

America/Rio_Branco	UTC05:00	UTC05:00	Acre
America/Rosario	UTC03:00	UTC03:00	
America/Santa_Isabel	UTC08:00	UTC07:00	
America/Santarem	UTC03:00	UTC03:00	Para (west)
America/Santiago	UTC04:00	UTC03:00	Chile (most areas)
America/Santo_Domingo	UTC04:00	UTC04:00	
America/Sao_Paulo	UTC03:00	UTC03:00	Brazil (southeast: GO, DF, MG, ES, RJ, SP, PR, SC, RS)
America/Scoresbysund	UTC01:00	UTC+00:00	Scoresbysund/Itoqqortoormiit
America/Shiprock	UTC07:00	UTC06:00	
America/Sitka	UTC09:00	UTC08:00	Alaska - Sitka area
America/St_Barthelemy	UTC04:00	UTC04:00	
America/St_Johns	UTC03:30	UTC02:30	Newfoundland; Labrador (southeast)
America/St_Kitts	UTC04:00	UTC04:00	
America/St_Lucia	UTC04:00	UTC04:00	
America/St_Thomas	UTC04:00	UTC04:00	
America/St_Vincent	UTC04:00	UTC04:00	
America/Swift_Current	UTC06:00	UTC06:00	CST - SK (midwest)
America/Tegucigalpa	UTC06:00	UTC06:00	
America/Thule	UTC04:00	UTC03:00	Thule/Pituffik
America/Thunder_Bay	UTC05:00	UTC04:00	Eastern - ON (Thunder Bay)
America/Tijuana	UTC08:00	UTC07:00	Pacific Time US - Baja California
America/Toronto	UTC05:00	UTC04:00	Eastern - ON, QC (most areas)
America/Tortola	UTC04:00	UTC04:00	
America/Vancouver	UTC08:00	UTC07:00	Pacific - BC (most areas)
America/Virgin	UTC04:00	UTC04:00	
America/Whitehorse	UTC08:00	UTC07:00	Pacific - Yukon (south)
America/Winnipeg	UTC06:00	UTC05:00	Central - ON (west); Manitoba
America/Yakutat	UTC09:00	UTC08:00	Alaska - Yakutat
America/Yellowknife	UTC07:00	UTC06:00	Mountain - NT (central)
Antarctica/Casey	UTC+11:00	UTC+11:00	Casey
Antarctica/Davis	UTC+07:00	UTC+07:00	Davis
Antarctica/DumontDURville	UTC+10:00	UTC+10:00	Dumont-d'Urville
Antarctica/Macquarie	UTC+11:00	UTC+11:00	Macquarie Island
Antarctica/Mawson	UTC+05:00	UTC+05:00	Mawson
Antarctica/McMurdo	UTC+12:00	UTC+13:00	New Zealand time - McMurdo, South Pole
Antarctica/Palmer	UTC03:00	UTC03:00	Palmer
Antarctica/Rothera	UTC03:00	UTC03:00	Rothera
Antarctica/South_Pole	UTC+12:00	UTC+13:00	
Antarctica/Syowa	UTC+03:00	UTC+03:00	Syowa
Antarctica/Troll	UTC+00:00	UTC+02:00	Troll

Antarctica/Vostok	UTC+06:00	UTC+06:00	Vostok
Arctic/Longyearbyen	UTC+01:00	UTC+02:00	
Asia/Aden	UTC+03:00	UTC+03:00	
Asia/Almaty	UTC+06:00	UTC+06:00	Kazakhstan (most areas)
Asia/Amman	UTC+02:00	UTC+03:00	
Asia/Anadyr	UTC+12:00	UTC+12:00	MSK+09 - Bering Sea
Asia/Aqtau	UTC+05:00	UTC+05:00	Mangghystau/Mankistau
Asia/Aqtobe	UTC+05:00	UTC+05:00	Aqtobe/Aktobe
Asia/Ashgabat	UTC+05:00	UTC+05:00	
Asia/Ashkhabad	UTC+05:00	UTC+05:00	
Asia/Atyrau	UTC+05:00	UTC+05:00	Atyrau/Atirau/Gur'yev
Asia/Baghdad	UTC+03:00	UTC+03:00	
Asia/Bahrain	UTC+03:00	UTC+03:00	
Asia/Baku	UTC+04:00	UTC+04:00	
Asia/Bangkok	UTC+07:00	UTC+07:00	
Asia/Barnaul	UTC+07:00	UTC+07:00	MSK+04 - Altai
Asia/Beirut	UTC+02:00	UTC+03:00	
Asia/Bishkek	UTC+06:00	UTC+06:00	
Asia/Brunei	UTC+08:00	UTC+08:00	
Asia/Calcutta	UTC+05:30	UTC+05:30	
Asia/Chita	UTC+09:00	UTC+09:00	MSK+06 - Zabaykalsky
Asia/Choibalsan	UTC+08:00	UTC+08:00	Dornod, Sukhbaatar
Asia/Chongqing	UTC+08:00	UTC+08:00	
Asia/Chungking	UTC+08:00	UTC+08:00	
Asia/Colombo	UTC+05:30	UTC+05:30	
Asia/Dacca	UTC+06:00	UTC+06:00	
Asia/Damascus	UTC+02:00	UTC+03:00	
Asia/Dhaka	UTC+06:00	UTC+06:00	
Asia/Dili	UTC+09:00	UTC+09:00	
Asia/Dubai	UTC+04:00	UTC+04:00	
Asia/Dushanbe	UTC+05:00	UTC+05:00	
Asia/Famagusta	UTC+02:00	UTC+02:00	Northern Cyprus
Asia/Gaza	UTC+02:00	UTC+03:00	Gaza Strip
Asia/Harbin	UTC+08:00	UTC+08:00	
Asia/Hebron	UTC+02:00	UTC+03:00	West Bank
Asia/Ho_Chi_Minh	UTC+07:00	UTC+07:00	
Asia/Hong_Kong	UTC+08:00	UTC+08:00	
Asia/Hovd	UTC+07:00	UTC+07:00	Bayan-Olgii, Govi-Altai, Hovd, Uvs, Zavkhan
Asia/Irkutsk	UTC+08:00	UTC+08:00	MSK+05 - Irkutsk, Buryatia
Asia/Istanbul	UTC+03:00	UTC+03:00	

Asia/Jakarta	UTC+07:00	UTC+07:00	Java, Sumatra
Asia/Jayapura	UTC+09:00	UTC+09:00	New Guinea (West Papua / Irian Jaya); Maluku/Moluccas
Asia/Jerusalem	UTC+02:00	UTC+03:00	
Asia/Kabul	UTC+04:30	UTC+04:30	
Asia/Kamchatka	UTC+12:00	UTC+12:00	MSK+09 - Kamchatka
Asia/Karachi	UTC+05:00	UTC+05:00	
Asia/Kashgar	UTC+06:00	UTC+06:00	
Asia/Kathmandu	UTC+05:45	UTC+05:45	
Asia/Katmandu	UTC+05:45	UTC+05:45	
Asia/Khandyga	UTC+09:00	UTC+09:00	MSK+06 - Tomponsky, Ust-Maysky
Asia/Kolkata	UTC+05:30	UTC+05:30	
Asia/Krasnoyarsk	UTC+07:00	UTC+07:00	MSK+04 - Krasnoyarsk area
Asia/Kuala_Lumpur	UTC+08:00	UTC+08:00	Malaysia (peninsula)
Asia/Kuching	UTC+08:00	UTC+08:00	Sabah, Sarawak
Asia/Kuwait	UTC+03:00	UTC+03:00	
Asia/Macao	UTC+08:00	UTC+08:00	
Asia/Macau	UTC+08:00	UTC+08:00	
Asia/Magadan	UTC+11:00	UTC+11:00	MSK+08 - Magadan
Asia/Makassar	UTC+08:00	UTC+08:00	Borneo (east, south); Sulawesi/Celebes, Bali, Nusa Tenggara; Timor (west)
Asia/Manila	UTC+08:00	UTC+08:00	
Asia/Muscat	UTC+04:00	UTC+04:00	
Asia/Novokuznetsk	UTC+07:00	UTC+07:00	MSK+04 - Kemerovo
Asia/Novosibirsk	UTC+07:00	UTC+07:00	MSK+04 - Novosibirsk
Asia/Omsk	UTC+06:00	UTC+06:00	MSK+03 - Omsk
Asia/Oral	UTC+05:00	UTC+05:00	West Kazakhstan
Asia/Phnom_Penh	UTC+07:00	UTC+07:00	
Asia/Pontianak	UTC+07:00	UTC+07:00	Borneo (west, central)
Asia/Pyongyang	UTC+09:00	UTC+09:00	
Asia/Qatar	UTC+03:00	UTC+03:00	
Asia/Qyzylorda	UTC+05:00	UTC+05:00	Qyzylorda/Kyzylorda/Kzyl-Orda
Asia/Rangoon	UTC+06:30	UTC+06:30	
Asia/Riyadh	UTC+03:00	UTC+03:00	
Asia/Saigon	UTC+07:00	UTC+07:00	
Asia/Sakhalin	UTC+11:00	UTC+11:00	MSK+08 - Sakhalin Island
Asia/Samarkand	UTC+05:00	UTC+05:00	Uzbekistan (west)
Asia/Seoul	UTC+09:00	UTC+09:00	
Asia/Shanghai	UTC+08:00	UTC+08:00	Beijing Time
Asia/Singapore	UTC+08:00	UTC+08:00	
Asia/Srednekolymsk	UTC+11:00	UTC+11:00	MSK+08 - Sakha (E); North Kuril Is
Asia/Taipei	UTC+08:00	UTC+08:00	

Asia/Tashkent	UTC+05:00	UTC+05:00	Uzbekistan (east)
Asia/Tbilisi	UTC+04:00	UTC+04:00	
Asia/Tehran	UTC+03:30	UTC+04:30	
Asia/Tel_Aviv	UTC+02:00	UTC+03:00	
Asia/Thimbu	UTC+06:00	UTC+06:00	
Asia/Thimphu	UTC+06:00	UTC+06:00	
Asia/Tokyo	UTC+09:00	UTC+09:00	
Asia/Tomsk	UTC+07:00	UTC+07:00	MSK+04 - Tomsk
Asia/Ujung_Pandang	UTC+08:00	UTC+08:00	
Asia/Ulaanbaatar	UTC+08:00	UTC+08:00	Mongolia (most areas)
Asia/Ulan_Bator	UTC+08:00	UTC+08:00	
Asia/Urumqi	UTC+06:00	UTC+06:00	Xinjiang Time
Asia/Ust-Nera	UTC+10:00	UTC+10:00	MSK+07 - Oymyakonsky
Asia/Vientiane	UTC+07:00	UTC+07:00	
Asia/Vladivostok	UTC+10:00	UTC+10:00	MSK+07 - Amur River
Asia/Yakutsk	UTC+09:00	UTC+09:00	MSK+06 - Lena River
Asia/Yangon	UTC+06:30	UTC+06:30	
Asia/Yekaterinburg	UTC+05:00	UTC+05:00	MSK+02 - Urals
Asia/Yerevan	UTC+04:00	UTC+04:00	
Atlantic/Azores	UTC01:00	UTC+00:00	Azores
Atlantic/Bermuda	UTC04:00	UTC03:00	
Atlantic/Canary	UTC+00:00	UTC+01:00	Canary Islands
Atlantic/Cape_Verde	UTC01:00	UTC01:00	
Atlantic/Faeroe	UTC+00:00	UTC+01:00	
Atlantic/Faroe	UTC+00:00	UTC+01:00	
Atlantic/Jan_Mayen	UTC+01:00	UTC+02:00	
Atlantic/Madeira	UTC+00:00	UTC+01:00	Madeira Islands
Atlantic/Reykjavik	UTC+00:00	UTC+00:00	
Atlantic/South_Georgia	UTC02:00	UTC02:00	
Atlantic/St_Helena	UTC+00:00	UTC+00:00	
Atlantic/Stanley	UTC03:00	UTC03:00	
Australia/ACT	UTC+10:00	UTC+11:00	
Australia/Adelaide	UTC+09:30	UTC+10:30	South Australia
Australia/Brisbane	UTC+10:00	UTC+10:00	Queensland (most areas)
Australia/Broken_Hill	UTC+09:30	UTC+10:30	New South Wales (Yancowinna)
Australia/Canberra	UTC+10:00	UTC+11:00	
Australia/Currie	UTC+10:00	UTC+11:00	Tasmania (King Island)
Australia/Darwin	UTC+09:30	UTC+09:30	Northern Territory
Australia/Eucla	UTC+08:45	UTC+08:45	Western Australia (Eucla)
Australia/Hobart	UTC+10:00	UTC+11:00	Tasmania (most areas)

Australia/LHI	UTC+10:30	UTC+11:00	
Australia/Lindeman	UTC+10:00	UTC+10:00	Queensland (Whitsunday Islands)
Australia/Lord_Howe	UTC+10:30	UTC+11:00	Lord Howe Island
Australia/Melbourne	UTC+10:00	UTC+11:00	Victoria
Australia/North	UTC+09:30	UTC+09:30	
Australia/NSW	UTC+10:00	UTC+11:00	
Australia/Perth	UTC+08:00	UTC+08:00	Western Australia (most areas)
Australia/Queensland	UTC+10:00	UTC+10:00	
Australia/South	UTC+09:30	UTC+10:30	
Australia/Sydney	UTC+10:00	UTC+11:00	New South Wales (most areas)
Australia/Tasmania	UTC+10:00	UTC+11:00	
Australia/Victoria	UTC+10:00	UTC+11:00	
Australia/West	UTC+08:00	UTC+08:00	
Australia/Yancowinna	UTC+09:30	UTC+10:30	
Brazil/Acre	UTC05:00	UTC05:00	
Brazil/DeNoronha	UTC02:00	UTC02:00	
Brazil/East	UTC03:00	UTC02:00	
Brazil/West	UTC04:00	UTC04:00	
Canada/Atlantic	UTC04:00	UTC03:00	
Canada/Central	UTC06:00	UTC05:00	
Canada/Eastern	UTC05:00	UTC04:00	
Canada/Mountain	UTC07:00	UTC06:00	
Canada/Newfoundland	UTC03:30	UTC02:30	
Canada/Pacific	UTC08:00	UTC07:00	
Canada/Saskatchewan	UTC06:00	UTC06:00	
Canada/Yukon	UTC08:00	UTC07:00	
CET	UTC+01:00	UTC+02:00	
Chile/Continental	UTC04:00	UTC03:00	
Chile/EasterIsland	UTC06:00	UTC05:00	
CST6CDT	UTC06:00	UTC05:00	
Cuba	UTC05:00	UTC04:00	
EET	UTC+02:00	UTC+03:00	
Egypt	UTC+02:00	UTC+02:00	
Eire	UTC+00:00	UTC+01:00	
EST	UTC05:00	UTC05:00	
EST5EDT	UTC05:00	UTC04:00	
Etc/GMT	UTC+00:00	UTC+00:00	
Etc/GMT+0	UTC+00:00	UTC+00:00	
Etc/GMT+1	UTC01:00	UTC01:00	
Etc/GMT+10	UTC10:00	UTC10:00	

Etc/GMT+11	UTC11:00	UTC11:00	
Etc/GMT+12	UTC12:00	UTC12:00	
Etc/GMT+2	UTC02:00	UTC02:00	
Etc/GMT+3	UTC03:00	UTC03:00	
Etc/GMT+4	UTC04:00	UTC04:00	
Etc/GMT+5	UTC05:00	UTC05:00	
Etc/GMT+6	UTC06:00	UTC06:00	
Etc/GMT+7	UTC07:00	UTC07:00	
Etc/GMT+8	UTC08:00	UTC08:00	
Etc/GMT+9	UTC09:00	UTC09:00	
Etc/GMT0	UTC+00:00	UTC+00:00	
Etc/GMT-0	UTC+00:00	UTC+00:00	
Etc/GMT-1	UTC+01:00	UTC+01:00	
Etc/GMT-10	UTC+10:00	UTC+10:00	
Etc/GMT-11	UTC+11:00	UTC+11:00	
Etc/GMT-12	UTC+12:00	UTC+12:00	
Etc/GMT-13	UTC+13:00	UTC+13:00	
Etc/GMT-14	UTC+14:00	UTC+14:00	
Etc/GMT-2	UTC+02:00	UTC+02:00	
Etc/GMT-3	UTC+03:00	UTC+03:00	
Etc/GMT-4	UTC+04:00	UTC+04:00	
Etc/GMT-5	UTC+05:00	UTC+05:00	
Etc/GMT-6	UTC+06:00	UTC+06:00	
Etc/GMT-7	UTC+07:00	UTC+07:00	
Etc/GMT-8	UTC+08:00	UTC+08:00	
Etc/GMT-9	UTC+09:00	UTC+09:00	
Etc/Greenwich	UTC+00:00	UTC+00:00	
Etc/UCT	UTC+00:00	UTC+00:00	
Etc/Universal	UTC+00:00	UTC+00:00	
Etc/UTC	UTC+00:00	UTC+00:00	
Etc/Zulu	UTC+00:00	UTC+00:00	
Europe/Amsterdam	UTC+01:00	UTC+02:00	
Europe/Andorra	UTC+01:00	UTC+02:00	
Europe/Astrakhan	UTC+04:00	UTC+04:00	MSK+01 - Astrakhan
Europe/Athens	UTC+02:00	UTC+03:00	
Europe/Belfast	UTC+00:00	UTC+01:00	
Europe/Belgrade	UTC+01:00	UTC+02:00	
Europe/Berlin	UTC+01:00	UTC+02:00	Germany (except for Büsingen am Hochrhein)
Europe/Bratislava	UTC+01:00	UTC+02:00	
Europe/Brussels	UTC+01:00	UTC+02:00	

Europe/Bucharest	UTC+02:00	UTC+03:00	
Europe/Budapest	UTC+01:00	UTC+02:00	
Europe/Busingen	UTC+01:00	UTC+02:00	Büdingen am Hochrhein
Europe/Chisinau	UTC+02:00	UTC+03:00	
Europe/Copenhagen	UTC+01:00	UTC+02:00	
Europe/Dublin	UTC+00:00	UTC+01:00	
Europe/Gibraltar	UTC+01:00	UTC+02:00	
Europe/Guernsey	UTC+00:00	UTC+01:00	
Europe/Helsinki	UTC+02:00	UTC+03:00	
Europe/Isle_of_Man	UTC+00:00	UTC+01:00	
Europe/Istanbul	UTC+03:00	UTC+03:00	
Europe/Jersey	UTC+00:00	UTC+01:00	
Europe/Kaliningrad	UTC+02:00	UTC+02:00	MSK01 - Kaliningrad
Europe/Kiev	UTC+02:00	UTC+03:00	Ukraine (most areas)
Europe/Kirov	UTC+03:00	UTC+03:00	MSK+00 - Kirov
Europe/Lisbon	UTC+00:00	UTC+01:00	Portugal (mainland)
Europe/Ljubljana	UTC+01:00	UTC+02:00	
Europe/London	UTC+00:00	UTC+01:00	
Europe/Luxembourg	UTC+01:00	UTC+02:00	
Europe/Madrid	UTC+01:00	UTC+02:00	Spain (mainland)
Europe/Malta	UTC+01:00	UTC+02:00	
Europe/Mariehamn	UTC+02:00	UTC+03:00	
Europe/Minsk	UTC+03:00	UTC+03:00	
Europe/Monaco	UTC+01:00	UTC+02:00	
Europe/Moscow	UTC+03:00	UTC+03:00	MSK+00 - Moscow area
Asia/Nicosia	UTC+02:00	UTC+03:00	Cyprus (most areas)
Europe/Oslo	UTC+01:00	UTC+02:00	
Europe/Paris	UTC+01:00	UTC+02:00	
Europe/Podgorica	UTC+01:00	UTC+02:00	
Europe/Prague	UTC+01:00	UTC+02:00	
Europe/Riga	UTC+02:00	UTC+03:00	
Europe/Rome	UTC+01:00	UTC+02:00	
Europe/Samara	UTC+04:00	UTC+04:00	MSK+01 - Samara, Udmurtia
Europe/San_Marino	UTC+01:00	UTC+02:00	
Europe/Sarajevo	UTC+01:00	UTC+02:00	
Europe/Saratov	UTC+04:00	UTC+04:00	MSK+01 - Saratov
Europe/Simferopol	UTC+03:00	UTC+03:00	Crimea
Europe/Skopje	UTC+01:00	UTC+02:00	
Europe/Sofia	UTC+02:00	UTC+03:00	
Europe/Stockholm	UTC+01:00	UTC+02:00	

Europe/Tallinn	UTC+02:00	UTC+03:00	
Europe/Tirane	UTC+01:00	UTC+02:00	
Europe/Tiraspol	UTC+02:00	UTC+03:00	
Europe/Ulyanovsk	UTC+04:00	UTC+04:00	MSK+01 - Ulyanovsk
Europe/Uzhgorod	UTC+02:00	UTC+03:00	Ruthenia
Europe/Vaduz	UTC+01:00	UTC+02:00	
Europe/Vatican	UTC+01:00	UTC+02:00	
Europe/Vienna	UTC+01:00	UTC+02:00	
Europe/Vilnius	UTC+02:00	UTC+03:00	
Europe/Volgograd	UTC+04:00	UTC+04:00	MSK+01 - Volgograd
Europe/Warsaw	UTC+01:00	UTC+02:00	
Europe/Zagreb	UTC+01:00	UTC+02:00	
Europe/Zaporozhye	UTC+02:00	UTC+03:00	Zaporozh'ye/Zaporizhia; Lugansk/Luhansk (east)
Europe/Zurich	UTC+01:00	UTC+02:00	
GB	UTC+00:00	UTC+01:00	
GB-Eire	UTC+00:00	UTC+01:00	
GMT	UTC+00:00	UTC+00:00	
GMT+0	UTC+00:00	UTC+00:00	
GMT0	UTC+00:00	UTC+00:00	
GMT-0	UTC+00:00	UTC+00:00	
Greenwich	UTC+00:00	UTC+00:00	
Hongkong	UTC+08:00	UTC+08:00	
HST	UTC10:00	UTC10:00	
Iceland	UTC+00:00	UTC+00:00	
Indian/Antananarivo	UTC+03:00	UTC+03:00	
Indian/Chagos	UTC+06:00	UTC+06:00	
Indian/Christmas	UTC+07:00	UTC+07:00	
Indian/Cocos	UTC+06:30	UTC+06:30	
Indian/Comoro	UTC+03:00	UTC+03:00	
Indian/Kerguelen	UTC+05:00	UTC+05:00	
Indian/Mahe	UTC+04:00	UTC+04:00	
Indian/Maldives	UTC+05:00	UTC+05:00	
Indian/Mauritius	UTC+04:00	UTC+04:00	
Indian/Mayotte	UTC+03:00	UTC+03:00	
Indian/Reunion	UTC+04:00	UTC+04:00	
Iran	UTC+03:30	UTC+04:30	
Israel	UTC+02:00	UTC+03:00	
Jamaica	UTC05:00	UTC05:00	
Japan	UTC+09:00	UTC+09:00	
Kwajalein	UTC+12:00	UTC+12:00	

Libya	UTC+02:00	UTC+02:00	
MET	UTC+01:00	UTC+02:00	
Mexico/BajaNorte	UTC08:00	UTC07:00	
Mexico/BajaSur	UTC07:00	UTC06:00	
Mexico/General	UTC06:00	UTC05:00	
MST	UTC07:00	UTC07:00	
MST7MDT	UTC07:00	UTC06:00	
Navajo	UTC07:00	UTC06:00	
NZ	UTC+12:00	UTC+13:00	
NZ-CHAT	UTC+12:45	UTC+13:45	
Pacific/Apia	UTC+13:00	UTC+14:00	
Pacific/Auckland	UTC+12:00	UTC+13:00	New Zealand (most areas)
Pacific/Bougainville	UTC+11:00	UTC+11:00	Bougainville
Pacific/Chatham	UTC+12:45	UTC+13:45	Chatham Islands
Pacific/Chuuk	UTC+10:00	UTC+10:00	Chuuk/Truk, Yap
Pacific/Easter	UTC06:00	UTC05:00	Easter Island
Pacific/Efate	UTC+11:00	UTC+11:00	
Pacific/Enderbury	UTC+13:00	UTC+13:00	Phoenix Islands
Pacific/Fakaofu	UTC+13:00	UTC+13:00	
Pacific/Fiji	UTC+12:00	UTC+13:00	
Pacific/Funafuti	UTC+12:00	UTC+12:00	
Pacific/Galapagos	UTC06:00	UTC06:00	Galapagos Islands
Pacific/Gambier	UTC09:00	UTC09:00	Gambier Islands
Pacific/Guadalcanal	UTC+11:00	UTC+11:00	
Pacific/Guam	UTC+10:00	UTC+10:00	
Pacific/Honolulu	UTC10:00	UTC10:00	Hawaii
Pacific/Johnston	UTC10:00	UTC10:00	
Pacific/Kiritimati	UTC+14:00	UTC+14:00	Line Islands
Pacific/Kosrae	UTC+11:00	UTC+11:00	Kosrae
Pacific/Kwajalein	UTC+12:00	UTC+12:00	Kwajalein
Pacific/Majuro	UTC+12:00	UTC+12:00	Marshall Islands (most areas)
Pacific/Marquesas	UTC09:30	UTC09:30	Marquesas Islands
Pacific/Midway	UTC11:00	UTC11:00	Midway Islands
Pacific/Nauru	UTC+12:00	UTC+12:00	
Pacific/Niue	UTC11:00	UTC11:00	
Pacific/Norfolk	UTC+11:00	UTC+11:00	
Pacific/Noumea	UTC+11:00	UTC+11:00	
Pacific/Pago_Pago	UTC11:00	UTC11:00	
Pacific/Palau	UTC+09:00	UTC+09:00	
Pacific/Pitcairn	UTC08:00	UTC08:00	

Pacific/Pohnpei	UTC+11:00	UTC+11:00	Pohnpei/Ponape
Pacific/Ponape	UTC+11:00	UTC+11:00	
Pacific/Port_Moresby	UTC+10:00	UTC+10:00	Papua New Guinea (most areas)
Pacific/Rarotonga	UTC10:00	UTC10:00	
Pacific/Saipan	UTC+10:00	UTC+10:00	
Pacific/Samoa	UTC11:00	UTC11:00	
Pacific/Tahiti	UTC10:00	UTC10:00	Society Islands
Pacific/Tarawa	UTC+12:00	UTC+12:00	Gilbert Islands
Pacific/Tongatapu	UTC+13:00	UTC+14:00	
Pacific/Truk	UTC+10:00	UTC+10:00	
Pacific/Wake	UTC+12:00	UTC+12:00	Wake Island
Pacific/Wallis	UTC+12:00	UTC+12:00	
Pacific/Yap	UTC+10:00	UTC+10:00	
Poland	UTC+01:00	UTC+02:00	
Portugal	UTC+00:00	UTC+01:00	
PRC	UTC+08:00	UTC+08:00	
PST8PDT	UTC08:00	UTC07:00	
ROC	UTC+08:00	UTC+08:00	
ROK	UTC+09:00	UTC+09:00	
Singapore	UTC+08:00	UTC+08:00	
Turkey	UTC+03:00	UTC+03:00	
UCT	UTC+00:00	UTC+00:00	
Universal	UTC+00:00	UTC+00:00	
US/Alaska	UTC09:00	UTC08:00	
US/Aleutian	UTC10:00	UTC09:00	
US/Arizona	UTC07:00	UTC07:00	
US/Central	UTC06:00	UTC05:00	
US/Eastern	UTC05:00	UTC04:00	
US/East-Indiana	UTC05:00	UTC04:00	
US/Hawaii	UTC10:00	UTC10:00	
US/Indiana-Starke	UTC06:00	UTC05:00	
US/Michigan	UTC05:00	UTC04:00	
US/Mountain	UTC07:00	UTC06:00	
US/Pacific	UTC08:00	UTC07:00	
US/Pacific-New	UTC08:00	UTC07:00	
US/Samoa	UTC11:00	UTC11:00	
UTC	UTC+00:00	UTC+00:00	
WET	UTC+00:00	UTC+01:00	
W-SU	UTC+03:00	UTC+03:00	
Zulu	UTC+00:00	UTC+00:00	

cron Schedule Syntax Reference

Contents:

- Overview of cron
 - Cron syntax
 - Special characters
- Examples
 - Hourly
 - Daily
 - Weekly
 - Weekdays
 - Monthly
 - Yearly
 - Other examples
 - Unsupported cron expressions
 - Invalid cron expressions

This section describes the syntax for defining scheduled executions using cron in the Designer Cloud powered by Trifacta® platform . Typically, this method is used for repeated schedules.

Flow schedules:

- Flow owners can define scheduled executions of flows from within the Flow View page.
- Collaborators can review and cannot edit schedules.

NOTE: Time zone settings defined in the Designer Cloud application page where you are specifying your cron schedule are used with the schedule. To use UTC time zone, select UTC in the drop-down.

Overview of cron

The Designer Cloud powered by Trifacta platform allows you to make use of cron, a widely used syntax, for specifying times that recur at regular intervals. You can use cron to specify schedules on a per-minute or annual basis and arbitrary intervals in between.

Cron syntax

A cron scheduled is defined as a space-separated string of values. The following cron example defines a schedule to be triggered at 11:30:00pm on February 1:

minute	hour	day of month	month	day of week
30	23	1	2	*

When all values are matched, the cron job is triggered.

NOTE: Specification of seconds is not supported.

Wildcards:

In the above cron expression, the wildcard * can be used to match any accepted value, which means that the cron value type is not a factor in determining this schedule. Since the wildcard is applied to the day of week value, the schedule can be triggered on any day of the week.

NOTE: You must use the * character in either the day-of-week or day-of-month fields. Specifying both fields in the same cron expression is not supported.

Legend:

Except for the final field (year), all fields are required in the cron expression. Special characters are described below the table.

Value	Type	Description	Supported Special Characters
30	minute	0-59	, - * /
23	hour	0-23	, - * /
1	day of month	1-31	, - * / L W
2	month	1-12	, - * /
*	day of week	0 - 6 or Sun - Sat 0, Sun, SUN = Sunday 1, Mon, MON = Monday ... 6, Sat, SAT = Saturday	, - * / L #
*	year	(Optional) You can specify year settings if needed. Default is *.	, - * /

Special characters

You can use the following special characters in your cron expressions.

Character	Description
*	("all values") - Wildcard to match all possible values in the field. For example, if you wanted your trigger to fire every minute of the 10pm hour, the minute character in the expression is *. An example is below.
-	Specify a range of values. For example, you could use 1 - 5 in the day-of-week field to match the work days of the week (Monday through Friday). An example is below.
,	Specify a discrete set of values. For example, an entry of 1 , 10 , 20 , 30 for the day of month field is triggered on the 1st, 10th, 20th, and 30th (if possible) of the month.
/	Specify increments of the field in the units of the field. For example, 5 / 20 in the minutes field matches on the 5th, 25th, and 45th minute of each hour.
L	Last value accepted in the range is accepted in the following fields: <ul style="list-style-type: none"> Day-of-month: Specifies the last day of the month for the currently selected month value. <ul style="list-style-type: none"> In January, this value matches with 31. In February, this value matches with 28 for non-leap years. In April, this value matches with 30. Day-of-week: <ul style="list-style-type: none"> By itself, it specifies the last day of the week, which matches with 6 (Saturday). When used with another value, it specifies the last matching value for the month. For example, 3L is the last Wednesday of the month.
W	Specifies the nearest matching weekday. For example, an entry of 22W in the day-of-month field matches on the nearest weekday to the 22nd of the month. If the 22nd is a Saturday, then the cron job matches on the 24th (the following Monday).

Tip: LW can be used in the day-of-month field to match on the last weekday of the month.

Specifies the nth day of the month. Examples for the day-of-week field:

- 3#4 - fourth Tuesday of the month
- 5#2 - second Thursday of the month

Examples

Below are some example cron schedules.

Hourly

Runs at minute 15 of every hour:

```
15 * * * *
```

Daily

Runs every day at 10pm:

```
0 22 * * *
```

Runs every minute of the 10pm hour every day:

```
* 22 * * *
```

Weekly

Runs every Tuesday at 3am:

```
0 3 * * 2
```

Weekdays

Runs each weekday at 8pm:

```
0 20 * * 1-5
```

Note that the above schedule runs at 10pm on Monday night and each night of the week at that time.

To refresh the flow for each weekday morning, you might choose to start the schedules on Sunday, in which the day-of-week value starts with 0 and ends with 4.

Monthly

Runs the first day of each month at 2:30am:

```
30 2 1 * *
```

Runs at 3:30pm on the nearest weekday (W) to the 25th of the month:

```
30 15 25W * *
```

- If the 25th is a Saturday, the above triggers on Friday the 24th.
- If the 25th is a Sunday, the above triggers on Monday the 26th.

Yearly

Runs at midnight of January 1 each year:

```
0 0 1 1 * *
```

Other examples

Expression	Meaning
<pre>0 12 * * *</pre>	Fire at 12pm (noon) every day
<pre>15 10 * * *</pre>	Fire at 10:15am every day
<pre>15 10 * * *</pre>	Fire at 10:15am every day
<pre>15 10 * * * *</pre>	Fire at 10:15am every day
<pre>15 10 * * * 2017</pre>	Fire at 10:15am every day during the year 2017
<pre>* 14 * * *</pre>	Fire every minute starting at 2pm and ending at 2:59pm, every day
<pre>0/5 14 * * *</pre>	Fire every 5 minutes starting at 2pm and ending at 2:55pm, every day
<pre>0/5 14,18 * * *</pre>	Fire every 5 minutes starting at 2pm and ending at 2:55pm, AND fire every 5 minutes starting at 6pm and ending at 6:55pm, every day
<pre>0-5 14 * * *</pre>	Fire every minute starting at 2pm and ending at 2:05pm, every day
<pre>10,44 14 * 3 WED</pre>	Fire at 2:10pm and at 2:44pm every Wednesday in the month of March.

15 10 * * MON-FRI	Fire at 10:15am every Monday, Tuesday, Wednesday, Thursday and Friday
15 10 15 * *	Fire at 10:15am on the 15th day of every month
15 10 L * *	Fire at 10:15am on the last day of every month
15 10 L-2 * *	Fire at 10:15am on the 2nd-to-last last day of every month
15 10 * * 5L	Fire at 10:15am on the last Friday of every month
15 10 * * 5L 2017-2019	Fire at 10:15am on every last friday of every month during the years 2017, 2018 and 2019
15 10 * * 5#3	Fire at 10:15am on the third Friday of every month
0 12 1/5 * *	Fire at 12pm (noon) every 5 days every month, starting on the first day of the month.
11 11 11 11 *	Fire every November 11th at 11:11am.

Unsupported cron expressions

NOTE: The Designer Cloud application does not support mixing / and – special characters in the same expressions.

Instead of expressing ranges in your cron syntax, you can reference all possible options.

Invalid expression	Valid expression
0 23 * 1-11/2 * *	0 23 * 2,4,6,8,10 * *

Invalid cron expressions

Expression	Meaning	Reason
15 10 * * * 2001	Fire at 10:15am every day during the year 2001	This cron expression is invalid because it will not generate any events in the future.

* * *

-

The cron expression should contain 6 or 7 fields.



Copyright © 2022 - Trifacta, Inc.
All rights reserved.