

Source: dremio-software-acceleration.md

Dremio Software - Acceleration

Source:
<https://docs.dremio.com/current/acceleration/>

Version: current [26.x]

On this page

Accelerate Queries

In Dremio, queries can be accelerated with Reflections and results cache.

Reflections

A Reflection is a precomputed and optimized copy of source data or a query result, designed to speed up query performance. It is derived from an existing table or view, known as its anchor. Reflections can be:

Autonomous: automatically created and managed by Dremio.

Manual: created and managed by you.

Dremio's query optimizer uses Reflections to accelerate queries by avoiding the need to scan the original data. Instead of querying the raw source, Dremio automatically rewrites queries to use Reflections when they provide the necessary results, without requiring you to reference them directly.

When Dremio receives a query, it determines first whether any Reflections have at least one table in common with the tables and views that the query references. If any Reflections do, Dremio evaluates them to determine whether they satisfy the query. Then, if any Reflections do satisfy the query, Dremio generates a query plan that uses them.

Dremio then compares the cost of the plan to the cost of executing the query directly against the tables, and selects the plan with the lower cost. Finally, Dremio executes the selected query plan. Typically, plans that use one or more Reflections are less expensive than plans that run against raw data.

Types

There are different types of Reflections tailored to specific workloads:

Raw Reflections: retain the same number of records as its anchor while allowing a subset of columns. It enhances query performance by materializing complex views, transforming data from non-performant sources into the Iceberg table format optimized

for large-scale analytics, and utilizing partitioning and sorting for faster access. By precomputing and storing data in an optimized format, raw Reflections significantly reduce query latency and improve overall efficiency.

Aggregation Reflections: accelerate BI-style queries that involve aggregations (GROUP BY queries) by precomputing results (like `SUM`, `COUNT`, `AVG`, `GROUP BY`) across selected dimensions and measures. By precomputing expensive computations, they significantly improve query performance at runtime. These Reflections are ideal for analytical workloads with frequent aggregations on large datasets.

External Reflections: reference precomputed tables in external data sources instead of materializing Reflections within Dremio, eliminating refresh overhead and storage costs. You can use an external Reflection by defining a view in Dremio that matches the precomputed table and map the view to the external data source. The data in the precomputed table is not refreshed by Dremio. When querying the view, Dremio’s query planner leverages the external Reflection to generate optimal execution plans, improving query performance without additional storage consumption in Dremio.

Starflake Reflections: optimize multi-table joins by leveraging precomputed relationships between fact and dimension tables. When joins do not duplicate rows, Dremio can accelerate queries using Reflections even if they include only a subset of the joins in Reflections, reducing the need for multiple Reflections on different combinations of tables.

Reflections Features and Data Format Compatibility Matrix

The following table outlines the availability of key Reflections features across supported data formats, including version-specific enhancements such as Autonomous Reflections, Live Reflections, and Intelligent Incremental Refresh.

Data Format	Autonomous Reflections	Automatic Raw Recommendation	Manual Reflections	Live Reflections (25.1+)	Automatic Raw Recommendation (25.0+)	Intelligent Incremental Refresh (24.3+)
Iceberg	Yes	Yes	Yes	Yes	Yes	Yes
UniForm	Yes	Yes	Yes	Yes	Yes	Yes
Parquet	Yes	Yes	Yes	No	Yes	Yes
Delta	No	No	Yes	No	No	No
Federated Sources	No	No	Yes	No	No	No

Results Cache

Results cache improves query performance by reusing results from previous executions of the same deterministic query, provided that the underlying dataset remains unchanged and the previous execution was by the same user. The results cache feature works out of the box, requires no configuration, and automatically caches and reuses results. Regardless of whether a query uses results cache, it always returns the same results.

Results cache is client-agnostic, meaning a query executed in the Dremio console will result in a cache hit even if it is later re-run through other clients like JDBC, ODBC, REST or Arrow Flight. For a query to use the cache, its query plan must remain identical to

the original cached version. Any changes to the schema or dataset generate a new query plan, invalidating the cache.

Results cache also supports seamless coordinator scale-out, allowing newly added coordinators to benefit immediately from previously cached results.

Cases Supported By Results Cache

Query result are cached in the following cases:

The SQL statement is a ``SELECT`` statement.

The query reads from an Iceberg, Parquet dataset, or from a raw Reflection defined on other Dremio supported data sources and formats, such as relational databases, ``CSV``, ``JSON``, or ``TEXT``.

The query does not contain dynamic functions such as ``QUERY_USER``, ``IS_MEMBER``, ``RAND``, ``CURRENT_DATE``, or ``NOW``.

The query does not reference ``SYS`` or ``INFORMATION_SCHEMA`` tables, or use external query.

The result set size, when stored in Arrow format, is less than or equal to 20 MB.

The query is not executed in Dremio console as a preview.

Viewing Whether Queries Used Results Cache

You can view the list of jobs on the Jobs page to determine if queries from data consumers were accelerated by the results cache.

To find whether a query was accelerated by a results cache:

Find the job that ran the query and look for the lightning-bolt icon next to it. This icon indicates that the query was accelerated using either Reflections or the results cache.

Click on the row representing the job that ran the query to view the job summary. The summary, displayed in the pane to the right, provides details on whether the query was accelerated using results cache or Reflections.



Storage

Cached results are stored in the distributed storage, configured in `dremio.conf`. Executors write cache entries as Arrow data files and read them when processing ``SELECT`` queries that result in a cache hit. Coordinators are responsible for managing the deletion of expired cache files.

Deletion

A background task running on one of the Dremio coordinators handles cache expiration. This task runs every hour to mark cache entries that have not been accessed in the

past 24 hours as expired and subsequently deletes them along with their associated cache files.

Considerations and Limitations

SQL queries executed through the Dremio console or a REST client that access the cache will rewrite the cached query results to the job results store to enable pagination. This behavior will be enhanced in future releases.

Additional Resources

Find out more about Reflections by enrolling in the [Data Product Fundamentals course in Dremio University](#).

Was this page helpful?

[Previous

DbVisualizer][current/client-applications/dbvisualizer-legacy)][Next

Autonomous Reflections][current/acceleration/autonomous-reflections)

Reflections

Types

Reflections Features and Data Format Compatibility Matrix

Results Cache

Cases Supported By Results Cache

Viewing Whether Queries Used Results Cache

Storage

Deletion

Considerations and Limitations

Additional Resources

Source:
<https://docs.dremio.com/current/acceleration/autonomous-reflections>

Version: current [26.x]

On this page

Autonomous Reflections Enterprise

Dremio automatically creates and drops [Reflections](#) based on query patterns to optimize performance for queries on Iceberg tables, Parquet tables, and views based on

them. These Reflections refresh in real time when Iceberg tables change or when Parquet table metadata is updated, turning Reflection management into an automated process.

Autonomous Reflections refresh automatically when source data changes on:

Iceberg tables: when the table is modified through Dremio or other engines. Dremio polls tables every 10 seconds.

Parquet datasets: when metadata is updated in Dremio.

Enable Autonomous Reflections

To enable Autonomous Reflections, follow these steps:

Navigate to **Settings**.

Go to the **Preferences** tab.

Toggle the **Autonomous Reflections** switch to enable the feature.

Once enabled, Dremio will automatically create and manage Reflections based on query workload analysis from the last seven days.

Set the Refresh Engine and Routing Rule for Autonomous Reflections

Dremio recommends configuring a dedicated refresh engine with at least two nodes to isolate refresh jobs associated with Autonomous Reflections. This isolation ensures that resource-intensive refresh jobs do not impact user workloads, preserving both query performance and refresh efficiency.

When you enable Autonomous Reflections in Dremio or Kubernetes deployments, you will be prompted to specify a refresh engine, and Dremio automatically creates routing rules to direct refresh jobs to the selected engine. For other deployments, you must manually create an engine or queue, depending on the deployment type, and define a routing rule `(query_type() = 'Autonomous Reflections')` at the top of the list to ensure refresh jobs for Autonomous Reflections are prioritized.

To replace the current refresh engine or queue for Autonomous Reflections, move the existing routing rule `(query_type() = 'Autonomous Reflections')` from the current engine to a new engine and, at the top of the list to set the highest priority. This ensures that refresh jobs are properly redirected to the newly designated engine.

If the assigned refresh engine reaches its capacity, Dremio Autonomous Reflections will pause. Users are then notified through the Dremio console, prompting them to scale up the refresh engine if necessary.

Behavior

Dremio creates up to 100 Reflections, with a maximum of 10 per day.

The actual number of Reflections depends on query patterns, as well as the configuration and utilization of the Dremio engine assigned to execute Reflection refreshes.

When Dremio determines that a Reflection has a low score, it is not immediately dropped. Instead:

The Reflection is disabled for 7 days before it is dropped.

Admins can then view disabled Reflections through the Autonomous Reflection History Log.

For Aggregation Reflections, Dremio creates system-managed views that cannot be modified or referenced by other views. Admin users can drop these views, but doing also deletes the associated Reflection.

If you disable the Autonomous Reflections feature, existing Reflections will continue to function normally, but Dremio will not add new Reflections or drop ineffective ones.

Limits

Autonomous Reflections are only used when fully synchronized with their source data to ensure up-to-date query results.

Autonomous Reflections cannot be modified and can only be dropped by Admins. When a Reflection is manually dropped, Dremio will not recreate it for the next 90 days.

View History Log for Autonomous Reflections

To view the history of changes to Autonomous Reflections in the last 30 days, follow these steps:

Navigate to **Settings**.

Select **Reflections** and click **History Log**.

Was this page helpful?

[Previous

Accelerate Queries](/current/acceleration/)[Next

Manually Manage Reflections](/current/acceleration/manual-reflections/)

Enable Autonomous Reflections

Set the Refresh Engine and Routing Rule for Autonomous Reflections

Behavior

Limits

View History Log for Autonomous Reflections

Source:
[https://docs.dremio.com/current/acceleration/m
anual-reflections/](https://docs.dremio.com/current/acceleration/manual-reflections/)

Version: current [26.x]

On this page

Manually Manage Reflections

With [Autonomous Reflections](#) reducing the need for manual work, you no longer need to create or manage Reflections. However, when Autonomous Reflections are not enabled or for situations that require manual control, this page provides guidance on getting Reflection recommendations, creating, managing, and testing Reflections in Dremio.

When a Reflection is created, Dremio runs a single job with two steps:

The first step writes the query results as a materialization to the distributed acceleration storage by running the ``REFRESH REFLECTION`` SQL command.

The second step registers the materialization table and its metadata with the catalog so that the query optimizer can find the Reflection's definition and structure.

The following screenshot shows the ``REFRESH REFLECTION`` SQL command used to create the Reflection named ``Super-duper reflection``:

[!Reflection creation job listed on the Jobs page in the Dremio console](#)

The Reflection creation job is listed as a single job on the Jobs page, as shown in the example below:

[!Reflection creation job listed on the Jobs page in the Dremio console](#)

Dremio creates all Reflections as Apache Iceberg tables, and the metadata for the Reflection is generated at the same time.

Reflection Recommendations

When [Autonomous Reflections](#) are not enabled, Dremio automatically provides recommendations to add and remove Reflections based on query patterns to optimize performance for queries on Iceberg tables, Parquet datasets, and views based on them. Recommendations to add Reflections are sorted by overall effectiveness, with the most effective recommendations shown on top. Effectiveness relates to metrics such as the estimated number of accelerated jobs, potential increase in query execution speedup, and potential time saved during querying. These are rough estimates based on past data that can give you insight into the potential benefits of each recommendation. Reflections created using these recommendations refresh automatically when source data changes on:

Iceberg tables: when the table is modified through Dremio or other engines. Dremio polls tables every 10 seconds.

Parquet datasets: when metadata is updated in Dremio.

To view and apply Reflection recommendations:

Navigate to Project Settings

Select Reflections.

Click **Reflections Recommendations** to access the list of suggested Reflections.

To apply a recommendation, click on the plus icon at the end of the corresponding row.

Reflections created using usage based recommendations are only used when fully synchronized with their source data to ensure up-to-date query results.

Manually Requesting Recommendations for Specific Jobs

You can generate recommendations for default raw and aggregation Reflections by submitting job IDs to accelerate specific SQL queries using either the ``SYS.RECOMMEND_REFLECTIONS`` table function or the [Recommendations API](#). Obtain the job IDs by looking them up on the [Jobs page](#).

Running the SYS.RECOMMEND_REFLECTIONS Table Function

The ``SYS.RECOMMEND_REFLECTIONS`` table function returns a table of one or more recommendations.

Syntax for the ``SYS.RECOMMEND_REFLECTIONS`` table function

```
SELECT * FROM TABLE(SYS.RECOMMEND_REFLECTIONS(ARRAY['<jobId>', '<jobId>']))
```

The ``SYS.RECOMMEND_REFLECTIONS`` function's argument must be an array literal.

Parameters

The ``SYS.RECOMMEND_REFLECTIONS`` table function has one parameter: an array that contains the job IDs for the SQL queries that you want to accelerate.

You can list up to 100 job IDs in each ``SYS.RECOMMEND_REFLECTIONS`` query. You must have permission to view every job you list in the query.

Output

The output for ``SYS.RECOMMEND_REFLECTIONS`` queries is a table that includes the following columns:

Column Name	Data Type	Description
view_sql	varchar	If the recommendation is for a default raw Reflection: The value in this field is <code>`Default Raw Reflection`</code> . If the recommendation is for an aggregation Reflection: An SQL command that creates a view based on the initial SQL query in the referenced job ID. Dremio uses this view to create an aggregation Reflection when you run the SQL command from the <code>`reflection_sql`</code> column. Before running this command, ensure that you have already created an Arctic catalog or folder named <code>`recommended_view`</code> .
reflection_sql	varchar	If the recommendation is for a default raw Reflection: An SQL command for defining the Reflection on an existing view. If the recommendation is for an aggregation Reflection: An SQL command that alters and adds an aggregation Reflection to the view that is created when you run the SQL command in the <code>`view_sql`</code> column. Before running this command, ensure that you

have run the corresponding command in the `view_sql` column. Dremio automatically consolidates recommendations for aggregation Reflections when possible to create a single recommendation for similar SQL queries, so the output table may not contain an individual row that corresponds to each job ID listed in the `SYS.RECOMMEND_REFLECTIONS` query.

| job_ids | list | The unique identifier of the jobs that ran the queries for which the recommendation is given.

| reflection_score | double | Dremio's score for the recommended Reflection's quality, on a scale of 0 (worst) to 100 (best). The reflection_score value considers the recommended Reflection's anticipated quality compared to existing Reflections and other recommended Reflections, as well as the likely improvement in query run times if the recommended Reflection is implemented.

| average_improvement_factor | double | The likely average multiplicative rate of improvement for each query if you implement the recommended Reflection. For example, if the average_improvement_factor value is 2.34, implementing the recommended Reflection is likely to speed up each query by 2.34 times, on average.

| average_improvement_ms | double | The likely average improvement, in milliseconds, for each query if you implement the recommended Reflection. For example, if the average_improvement_ms value is 5400, implementing the recommended Reflection is likely to save an average of 5400 milliseconds for each query that uses the Reflection.

Example Query and Output

To request a recommendation, you must have the job ID for at least one SQL query that you want to accelerate. For example, suppose this is the SQL query:

Example SQL Query

```
SELECT Max("p_size")
FROM s3.bucket1.tpch.sf10.parquet.part
JOIN s3.bucket1.tpch.sf10.parquet.lineitem
ON "p_partkey" = "l_partkey"
GROUP BY "p_brand"
```

Obtaining Recommendations

Retrieve the job ID for the query.

Run the `SYS.RECOMMEND_REFLECTIONS` query with the job ID. In the following example, the job ID is `844c0023-6272-8b16-aef3-aea289acadb1`:

Example `SYS.RECOMMEND_REFLECTIONS` Query

```
SELECT * FROM
TABLE(SYS.RECOMMEND_REFLECTIONS(ARRAY['844c0023-6272-8b16-aef3-aea289acadb1']))
```

The result of the query is returned in this table:

Example Results

| view_sql | reflection_sql | job_ids | reflection_score | average_improvement_factor |

```

average\improvement\ms |
| --- | --- | --- | --- | --- | --- |
|
|                                     CREATE                                     VIEW
"recommended_view"."Dataset\_9d74a03b-747a-42a2-a5ca-7f9c6f77b55d" AS SELECT
"part"."P\_BRAND" AS "F0[P\_BRAND]", "part"."P\_SIZE" AS "F1[P\_SIZE]",
"part"."P\_PARTKEY" AS "extra#0", "part"."P\_NAME" AS "extra#1", "part"."P\_MFGR" AS
"extra#2", "part"."P\_TYPE" AS "extra#4", "part"."P\_CONTAINER" AS "extra#6",
"part"."P\_RETAILPRICE" AS "extra#7", "part"."P\_COMMENT" AS "extra#8",
"lineitem"."L\_ORDERKEY" AS "extra#9", "lineitem"."L\_PARTKEY" AS "extra#10",
"lineitem"."L\_SUPPKEY" AS "extra#11", "lineitem"."L\_LINENUMBER" AS "extra#12",
"lineitem"."L\_QUANTITY" AS "extra#13", "lineitem"."L\_EXTENDEDPRICE" AS
"extra#14", "lineitem"."L\_DISCOUNT" AS "extra#15", "lineitem"."L\_TAX" AS
"extra#16", "lineitem"."L\_RETURNFLAG" AS "extra#17", "lineitem"."L\_LINESTATUS" AS
"extra#18", "lineitem"."L\_SHIPDATE" AS "extra#19", "lineitem"."L\_COMMITDATE" AS
"extra#20", "lineitem"."L\_RECEIPTDATE" AS "extra#21", "lineitem"."L\_SHIPINSTRUCT"
AS "extra#22", "lineitem"."L\_SHIPMODE" AS "extra#23", "lineitem"."L\_COMMENT" AS
"extra#24" FROM "s3"."bucket1"."tpch"."sf10"."parquet"."lineitem" INNER JOIN
"s3"."bucket1"."tpch"."sf10"."parquet"."part" ON "part"."P\_PARTKEY" =
"lineitem"."L\_PARTKEY" |
|                                     ALTER                                     DATASET
"recommended_view"."Dataset\_9d74a03b-747a-42a2-a5ca-7f9c6f77b55d" CREATE
AGGREGATE Reflection "agg\_0e07a376-7f8e-4c68-b2ce-6f6e819bebe6" USING
DIMENSIONS ("F0[P\_BRAND]") MEASURES ("F1[P\_SIZE]" (MAX)) |
"6j6c34cf-9drf-b07a-5ab7-abea69a66d00", "1a3c67c0-aab0-f9fb-97b4-af374b520100",
"1a3c67c0-db35-3645-9ef1-2a84e4d0ce00" | 50.00 | 10.00 | 7196 |

```

Using the Recommendation

If a recommendation is for an aggregation Reflection:

Create an Arctic catalog or folder named `recommended_view`.

Run the query listed in the `view_sql` column to create a view based on the initial SQL query.

Run the query listed in the `reflection_sql` column to create the recommended Reflection and wait for the Reflection to finish refreshing.

When the Reflection refresh is complete, run the SQL query to observe the acceleration due to the added Reflection.

Limitations

The `SYS.RECOMMEND_REFLECTIONS` table function has the following limitations:

The function cannot provide recommendations for jobs with multiple nested queries that contain joins.

An SQL query can contain only inner joins. Outer joins that are part of a view definition are also supported. Other types of joins are not supported.

An SQL query cannot contain Window functions.

Sending Requests to the Recommendations API

You can use the [Recommendations API](#) to submit the job IDs of jobs that ran SQL queries and receive recommendations for Reflections that can accelerate those queries.

Locations of the Reflections Editor

You use the Reflections editor to create, edit, and remove raw and aggregation Reflections.

If you know which table or view you want to create a Reflection from, follow either of these steps to open the Reflections editor:

In the space in which the anchor is located, click the gear icon in the Action field for the anchor. In the sidebar of the Dataset Settings window, select Reflections.

In the space in which the anchor is located, click the name of the anchor. Then, click Reflections in the bar at the top of the screen.

You can also open the Reflections editor from an existing Reflection that is listed in the Reflections page. Follow these steps to open the Reflections editor:

Open the Reflections page.

Click the gear in the sidebar, and then select Reflections in the sidebar of the Settings page.

Click the name of the Reflection. The Acceleration window is opened. The editor appears in this window.

Creating Raw Reflections

You can use the Reflections editor to create two types of raw Reflection:

A default raw Reflection that includes all of the columns of the anchor, but does not sort or horizontally partition on any columns

A raw Reflection that includes all or a subset of the columns of the anchor, and that does one or both of the following things:

Sorts on one or more columns

Horizontally partitions the data according to the values in one or more columns

note

For creating Reflections on views and tables with row-access and column-masking policies, see [Using Reflections on Datasets with Policies](#).

Prerequisites

If you want to accelerate queries on unoptimized data or data in slow storage, create a view that is itself created from a table in a non-columnar format or on slow-scan storage. You can then create your raw Reflection from that view.

If you want to accelerate "needle-in-a-haystack" queries, create a view that includes a

predicate to include only the rows that you want to scan. You can then create your raw Reflection from that view.

If you want to accelerate queries that perform expensive transformations, create a view that performs those transformations. You can then create your raw Reflection from that view.

If you want to accelerate queries that perform joins, create a view that performs the joins. You can then create your raw Reflection from that view.

Creating Default Raw Reflections

In the **Basic** view of the Reflections editor, you can create a raw Reflection that includes all of the fields that are in a table or view. Creating a basic raw Reflection ensures that Dremio never runs user queries against the underlying table or view when the raw Reflection is enabled.

Restrictions of the **Basic** View

You cannot select fields to sort or create horizontal partitions on.

The name of the Reflection that you create is restricted to "Raw Reflection".

You can create only one raw Reflection. If you want to create multiple raw Reflections at a time, use the **Advanced** view.

To create a raw Reflection in the **Basic** view of the Reflections editor:

Open the Reflections editor.

See Locations of the Reflections Editor to find out where you can open the editor from.

Click the toggle switch on the left side of the **Raw Reflections** bar.

Click **Save**.

tip

You can also create raw Reflections using [SQL commands](#).

For tips on what to do now after your raw Reflection is created and enabled, see [What to Do Next](#).

Creating Customized Raw Reflections

In the **Advanced** view of the Reflections editor, you can create one or more raw Reflections that include all or a selection of the fields that are in the anchor or supported anchor. You can also choose sort fields and fields for partitioning horizontally.

Dremio recommends that you follow the best practices listed in [Best Practices for Creating Raw and Aggregation Reflections](#) when you create customized raw Reflections.

If you make any of the following changes to a raw Reflection when you are using the **Advanced** view, you cannot switch to the **Basic** view:

Deselect one or more fields in the **Display** column. By default, all of the fields are selected.

Select one or more fields in the **Sort**, **Partition**, or **Distribute** column.

To create a raw Reflection in the **Advanced** view of the Reflections editor:

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

If the **Advanced** view is not already displayed, click the **Advanced View** button in the top-right corner of the editor.

Click the toggle switch in the table labeled **Raw Reflection** to enable the raw Reflection.

Queries do not start using the Reflection, however, until after you finished editing the Reflection and click **Save** in a later step.

![[/images/cloud/reflections-editor-raw-advanced-disabled.png)]]

(Optional) Click in the label to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Display** column to include fields in or exclude them from your Reflection.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in [Best Practices for Creating Raw and Aggregation Reflections](#).

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, recommended partition columns and transforms are selected for you. If you change the selection of columns, then this icon appears at the top of the table: ![This is the Recommendations icon](#). You can click it to revert back to the recommended selection of partition columns.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

- a. Click the gear icon in the table in which you are defining the Reflection.
- b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

What to Do Next

After you create a raw Reflection that is enabled, test whether the query optimizer is making queries use it. See [Testing Reflections](#) for the steps.

When you are sure that the Reflection is being used, follow one of these steps:

If the base table is in the Apache Iceberg format or the Delta Lake format: Set the schedule according to which all Reflections on the table are refreshed.

For all other base tables: Set the refresh type for all Reflections on the underlying table and set the schedule according to which they are refreshed.

See [Refreshing Reflections](#).

Creating Aggregation Reflections

Aggregation Reflections are summarized representations of data. Most BI tools generate aggregation and GROUP BY queries. Aggregation Reflections optimize these kinds of query patterns.

When you create aggregation Reflections, keep in mind these best practices:

Use dimensions that have relatively low cardinality in a table or view. The higher the cardinality of a dimension, the less benefit an aggregation Reflection has on query performance. Lower cardinality aggregation Reflections require less time to scan.

For a single table or view, create one aggregation Reflection for each important subset of dimensions in your queries, rather than one aggregation Reflection that includes all dimensions. Multiple small aggregation Reflections (versus one large one) are good for isolated pockets of query patterns on the same table or view that do not overlap. If your query patterns overlap, use fewer larger aggregation Reflections.

There are two cautions that accompany this advice, however:

Be careful of creating aggregation Reflections that have too few dimensions for your queries.

If a query uses more dimensions than are included in an aggregation Reflection, the Reflection cannot satisfy the query and the query optimizer does not run the query against it.

Be careful of creating more aggregation Reflections than are necessary to satisfy queries against a table or view.

The more Reflections you create, the more time the query optimizer requires to plan the execution of queries. Therefore, creating more aggregation Reflections than you need can slow down query performance, even if your aggregation Reflections are low-cardinality.

If you want to include a computed measure, first create a view with the computed column to use as a measure, and then create the aggregation Reflection on the view.

Dremio recommends that you also follow the best practices listed in [Best Practices for Creating Raw and Aggregation Reflections](#) when you create customized aggregation Reflections.

Creating Default Aggregation Reflections

You can use the **Basic** view of the Reflections editor to create one aggregation Reflection that includes fields, from the anchor or supported anchor, that are recommended for use as dimensions or measures. You can add or remove dimensions and measures, too.

Restrictions

You can create only one aggregation Reflection in the **Basic** view. If you want to create multiple aggregations Reflections at a time, use the **Advanced** view.

You cannot select fields for sorting or horizontally partitioning.

The name of the Reflection is restricted to "Aggregation Reflection".

Procedure

To create an aggregation Reflection in the **Basic** view of the Reflections editor:

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

In the **Aggregations Reflections** section of the editor, click **Generate** to get recommended fields to use as dimensions and measures. This will override any previously selected dimensions and measures. If you wish to proceed, click **Continue** in the confirmation dialog that follows.

In the **Aggregation Reflection** section of the editor, modify or accept the recommended fields for dimensions and measures.

To make the Reflection available to the query optimizer after you create it, click the toggle switch on the left side of the **Aggregation Reflections** bar.

Click **Save**.

tip

You can also create aggregation Reflections using [SQL commands](#).

For tips on what to do now after your aggregation Reflection is created and enabled, see [What to Do Next](#).

Creating Customized Aggregation Reflections

You can use the **Advanced** view of the Reflections editor to create one or more aggregation Reflections that select which fields in the anchor or supporting anchor to use as dimensions and measures. For each field that you use as a measure, you can use one or more of these SQL functions: APPROX_DISTINCT_COUNT, COUNT, MAX, and MIN. You can also choose sort fields and fields for partitioning horizontally.

Procedure

To create an aggregation Reflection in the **Advanced** view of the Reflections editor:

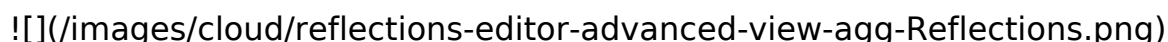
Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

Click the **Advanced View** button in the top-right corner of the editor.

Click **Aggregation Reflections**.

The Aggregation Reflections section is displayed, and one table for refining the aggregation Reflection that appeared in the **Basic** view is ready.



(Optional) Click in the name to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Dimensions** column to include or exclude fields to use as dimensions.

Click in the Measures field to include or exclude fields to use as measures. You can use one or more of these SQL functions for each measure: APPROX_DISTINCT_COUNT, COUNT, MAX, and MIN.


The full list of SQL aggregation functions that Dremio supports is not supported in the Reflections editor. If you want to create a Reflection that aggregates data by using the SQL function AVG, CORR, HLL, SUM, VAR_POP, or VAR_SAMP, you must create a view

that uses the function, and then create a raw Reflection from that view.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in Best Practices for Creating Raw and Aggregation Reflections.

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, recommended partition columns and transforms are selected for you. If you change the selection of columns, then this icon appears at the top of the table: . You can click it to revert back to the recommended selection of partition columns.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

- a. Click the gear icon in the table in which you are defining the Reflection.
- b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

What to Do Next

After you create an aggregation Reflection that is enabled, test whether the query optimizer is making queries use it. See [Testing Reflections](#) for the steps.

When you are sure that the Reflection is being used, follow one of these steps:

If the base table is in the Apache Iceberg format or the Delta Lake format: Set the schedule according to which all Reflections on the table are refreshed.

For all other base tables: Set the refresh type for all Reflections on the underlying table and set the schedule according to which they are refreshed.

See [Refreshing Reflections](#).

Editing Raw Reflections

You can edit an existing raw Reflection. You might want to do so if you are iteratively designing and testing a raw Reflection, if the definition of the view that the Reflection

was created from was changed, or if the schema of the underlying table was changed.

If you created a raw Reflection in the **Basic** view of the Reflections editor, you must use the **Advanced** view to edit it.

Dremio runs the job or jobs to recreate the Reflection after you click **Save**.

Procedure

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

Click the **Advanced View** button in the top-right corner of the editor.

In the **Raw Reflections** section of the **Advanced** view, locate the table that shows the definition of your Reflection.

(Optional) Click in the label to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Display** column to include fields in or exclude them from your Reflection.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in [Best Practices for Creating Raw and Aggregation Reflections](#).

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, partition columns and transforms are recommended for you. Hover over the [!This is the Recommendations icon.](#) icon at the top of the table to see the recommendation. Click the icon to accept the recommendation.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

a. Click the gear icon in the table in which you are defining the Reflection.

b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as

fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

Editing Aggregation Reflections

You might want to edit an aggregation Reflection if you are iteratively designing and testing an aggregation Reflection, if the definition of the view that the Reflection was created from was changed, if the schema of the underlying table was changed, or if you want to revise one or more aggregations defined in the Reflection.

If you created an aggregation Reflection in the **Basic** view of the Reflections editor, you can edit that Reflection either in the **Basic** view or in the **Advanced** view.

Dremio runs the job or jobs to recreate the Reflection after you click **Save**.

Editing Aggregation Reflections in the **Basic** View of the Reflections Editor

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

In the Aggregation Reflection section of the editor, modify or accept the recommendation for dimension fields and measure fields.

Click **Save**.

Editing Aggregation Reflections in the **Advanced** View of the Reflections Editor

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

Click the **Advanced View** button in the top-right corner of the editor.

Click **Aggregation Reflections**.

(Optional) Click in the name to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Dimensions** column to include or exclude fields to use as dimensions.

Click in the Measures field to include or exclude fields to use as measures. You can use one or more of these SQL functions for each measure: APPROX_DISTINCT_COUNT, COUNT, MAX, and MIN.

The full list of SQL aggregation functions that Dremio supports is not supported in the Reflections editor. If you want to create a Reflection that aggregates data by using the SQL function AVG, CORR, HLL, SUM, VAR_POP, or VAR_SAMP, you must create a view that uses the function, and then create a raw Reflection from that view.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in Best Practices for Creating Raw and Aggregation Reflections.

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, partition columns and transforms are recommended for you. Hover over the [!This is the Recommendations icon.](#) icon at the top of the table to see the recommendation. Click the icon to accept the recommendation.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

- a. Click the gear icon in the table in which you are defining the Reflection.
- b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

Creating External Reflections

See [External Reflections](#) for a description of what external Reflections are and their benefits.

To create an external Reflection:

Follow these steps in the data source:

- a. Select your source table.

b. Create a table that is derived from the source table, such as an aggregation table, if you do not already have one.

Follow these steps in Dremio:

a. Define a view on the derived table in the data source. The definition must match that of the derived table.

b. Define a new external Reflection that maps the view to the derived table.

note

The data types and column names in the external Reflection must match those in the view that the external Reflection is mapped to.

Example

Suppose you have a data source named `mySource` that is connected to Dremio. In that data source, there are (among all of your other tables) these two tables:

`sales`, which is a very large table of sales data.

`sales_by_region`, which aggregates by region the data that is in `sales`.

You want to make the data in `sales_by_region` available to data analysts who use Dremio. However, because you already have the `sales_by_region` table created, you do not see the need to create a Dremio table from `sales`, then create a Dremio view that duplicates `sales_by_region`, and finally create a Reflection on the view. You would like instead to make `sales_by_region` available to queries run from Bi tools through Dremio.

To do that, you follow these steps:

Create a view in Dremio that has the same definition as `sales_by_region`. Notice that the `FROM` clause points to the `sales` table that is in your data source, not to a Dremio table.

Example View

```
CREATE VIEW "myWorkspace"."sales_by_region" AS
SELECT
    AVG(sales_amount) average_sales,
    SUM(sales_amount) total_sales,
    COUNT(*) sales_count,
    region
FROM mySource.sales
GROUP BY region
```

Create an external Reflection that maps the view above to `sales_by_region` in `mySource`.

Example External Reflection

```
ALTER DATASET "myWorkspace"."sales_by_region"
CREATE EXTERNAL Reflection "external_sales_by_region"
USING "mySource"."sales_by_region"
```

The external Reflection lets Dremio's query planner know that there is a table in ``mySource`` that matches the Dremio view ``myWorkplace.sales_by_region`` and that can be used to satisfy queries against the view. When Dremio users query ``myWorkspace.sales_by_region``, Dremio routes the query to the data source ``mySource``, which runs the query against ``mySource.sales_by_region``.

Editing External Reflections

If you have modified the DDL of a derived table in your data source, follow these steps in Dremio to update the corresponding external Reflection:

Replace the view with one that has a definition that matches the definition of the derived table. When you do so, the external Reflection is dropped.

Define a new external Reflection that maps the view to the derived table.

Viewing Whether Queries Used Reflections

You can view the list of jobs on the Jobs page to find out whether queries were accelerated by Reflections. The Jobs page lists the jobs that ran queries, both queries from your data consumers and queries run within the Dremio user interface.

To find whether a query used a Reflection:

Find the job that ran the query by looking below the details in each row.

Look for the indicator that one or more Reflections were used. A lightning-bolt icon appears next to the job to indicate that a query was accelerated.

View the job summary by clicking the row that represents the job that ran the query. The job summary appears in the pane to the right of the list of jobs.

Relationship between Reflections and Jobs

The relationship between a job and a Reflection can be one of the following types:

CONSIDERED: the Reflection is defined on a dataset that is used in the query but was determined to not cover the query (for example the Reflection did not have a field that is used by the query).

MATCHED: a Reflection could have been used to accelerate the query but Dremio determined that it would not provide any benefits or another Reflection was determined to be a better choice.

CHOSEN: a Reflection is used to accelerate the query. Note that multiple Reflections can be used to accelerate queries.

Testing Reflections

You can test whether Reflections that you created are used to satisfy a query without actually running the query. This practice can be helpful when the tables are very large and you want to avoid processing large queries unnecessarily.

To test whether one or more Reflections are used by a query:

In the side navigation bar, click the SQL Runner icon.

In the SQL editor, type ``EXPLAIN PLAN FOR`` and then type or paste in your query.

Click the **Run** button.

When the query has finished, click the **Run** link found directly above the query results to view the job details. Any Reflections used will be shown on the page.

Setting the Expiration Policy for Reflections

Rather than delete a Reflection manually, you can specify how long you want Dremio to retain the Reflection before deleting it automatically.

note

Dremio does not allow expiration policies to be set on external Reflections or Reflections that automatically refresh when Iceberg data changes according to the [refresh policy](#).

To set the expiration policy for all Reflections derived from tables in a data source:

Right-click a data lake or external source.

Select **Edit Details**.

In the sidebar of the Edit Source window, click **Reflection Refresh**.

After making your changes, click **Save**. The changes take effect on the next refresh.

To set the expiration policy on Reflections derived from a particular table:

note

The table must be based on more than one file.

Locate a table.

Click the gear icon to its right.

In the sidebar of the Dataset Settings window, click **Reflection Refresh**.

After making your changes, click **Save**. The changes take effect on the next refresh.

Removing Reflections

You can choose to disable or delete Reflections.

Disabling Reflections

Disabled Reflections become unavailable for use by queries and will not be refreshed manually or according to their schedule.

note

Dremio does not disable external Reflections.

To disable a Reflection:

Open the Reflections editor for the Reflection's anchor or supporting anchor. See [Locations of the Reflections Editor](#).

Follow one of these steps:

If there is only one raw Reflection for the table or view, in the **Basic** view click the toggle switch in the **Raw Reflections** bar.

If there are two or more raw Reflections for the table or view, in the **Advanced** view click the toggle switch for the individual raw Reflection that you want to disable.

If there is only one aggregation Reflection for the table or view, in the **Basic** view click the toggle switch in the **Raw Reflections** bar.

If there are two or more aggregation Reflections for the table or view, in the **Advanced** view click the toggle switch for the individual aggregation Reflection that you want to disable.

Click **Save**. The changes take effect immediately.

Deleting Reflections

You can delete Reflections individually, or all of the Reflections on a table or view. When you delete a Reflection, its definition, data, and metadata are entirely deleted.

To delete a single raw or aggregation Reflection:

Open the Reflections editor for the Reflection's anchor or supporting anchor. See [Locations of the Reflections Editor](#).

Open the **Advanced** view, if it is not already open.

If the Reflection is an aggregation Reflection, click **Aggregation Reflections**.

Click the trash can icon for the Reflection that you want to delete.

Click **Save**. The deletion takes effect immediately.

To delete all raw and aggregation Reflections on a table or view:

Open the Reflections editor for the Reflection's anchor or supporting anchor. See [Locations of the Reflections Editor](#).

Click **Remove All Reflections**.

Click **Save**.

To delete an external Reflection, or to delete a raw or aggregation Reflection without using the Reflections editor, run this SQL command:

Delete a Reflection


```
ALTER DATASET <DATASET_PATH> DROP Reflection <REFLECTION_NAME>
```

DATASET_PATH: The path of the view on which the external Reflection is based.

REFLECTION_NAME: The name of the external Reflection.

Additional Resources

Find out more about Reflections by enrolling in the [Data Reflections Deep Dive course in Dremio University](#).

Was this page helpful?

[Previous

Autonomous Reflections](/current/acceleration/autonomous-reflections)[Next

View	Whether	Queries	Used
Reflections](/current/acceleration/manual-reflections/info-about-queries)			

Reflection Recommendations

Sending Requests to the Recommendations API

Locations of the Reflections Editor

Creating Raw Reflections

Prerequisites

Creating Default Raw Reflections

Creating Customized Raw Reflections

What to Do Next

Creating Aggregation Reflections

Creating Default Aggregation Reflections

Creating Customized Aggregation Reflections

What to Do Next

Editing Raw Reflections

Procedure

Editing Aggregation Reflections

Editing Aggregation Reflections in the **Basic** View of the Reflections Editor

Editing Aggregation Reflections in the **Advanced** View of the Reflections Editor

Creating External Reflections

Example

Editing External Reflections

Viewing Whether Queries Used Reflections

Relationship between Reflections and Jobs

Testing Reflections

Setting the Expiration Policy for Reflections

Removing Reflections

Disabling Reflections

Deleting Reflections

Additional Resources

Source:
[https://docs.dremio.com/current/acceleration/m](https://docs.dremio.com/current/acceleration/manual-reflections)
anual-reflections

Version: current [26.x]

On this page

Manually Manage Reflections

With Autonomous Reflections reducing the need for manual work, you no longer need to create or manage Reflections. However, when Autonomous Reflections are not enabled or for situations that require manual control, this page provides guidance on getting Reflection recommendations, creating, managing, and testing Reflections in Dremio.

When a Reflection is created, Dremio runs a single job with two steps:

The first step writes the query results as a materialization to the distributed acceleration storage by running the ``REFRESH REFLECTION`` SQL command.

The second step registers the materialization table and its metadata with the catalog so that the query optimizer can find the Reflection's definition and structure.

The following screenshot shows the ``REFRESH REFLECTION`` SQL command used to create the Reflection named ``Super-duper reflection``:

!Reflection creation job listed on the Jobs page in the Dremio console

The Reflection creation job is listed as a single job on the Jobs page, as shown in the example below:

!Reflection creation job listed on the Jobs page in the Dremio console

Dremio creates all Reflections as Apache Iceberg tables, and the metadata for the Reflection is generated at the same time.

Reflection Recommendations

When [Autonomous Reflections](#) are not enabled, Dremio automatically provides recommendations to add and remove Reflections based on query patterns to optimize performance for queries on Iceberg tables, Parquet datasets, and views based on them. Recommendations to add Reflections are sorted by overall effectiveness, with the most effective recommendations shown on top. Effectiveness relates to metrics such as the estimated number of accelerated jobs, potential increase in query execution speedup, and potential time saved during querying. These are rough estimates based on past data that can give you insight into the potential benefits of each recommendation. Reflections created using these recommendations refresh automatically when source data changes on:

Iceberg tables: when the table is modified through Dremio or other engines. Dremio polls tables every 10 seconds.

Parquet datasets: when metadata is updated in Dremio.
To view and apply Reflection recommendations:

Navigate to Project Settings

Select Reflections.

Click **Reflections Recommendations** to access the list of suggested Reflections.

To apply a recommendation, click on the plus icon at the end of the corresponding row.

Reflections created using usage based recommendations are only used when fully synchronized with their source data to ensure up-to-date query results.

Manually Requesting Recommendations for Specific Jobs

You can generate recommendations for default raw and aggregation Reflections by submitting job IDs to accelerate specific SQL queries using either the ``SYS.RECOMMEND_REFLECTIONS`` table function or the [Recommendations API](#). Obtain the job IDs by looking them up on the [Jobs page](#).

Running the SYS.RECOMMEND_REFLECTIONS Table Function

The ``SYS.RECOMMEND_REFLECTIONS`` table function returns a table of one or more recommendations.

Syntax for the ``SYS.RECOMMEND_REFLECTIONS`` table function

```
SELECT * FROM TABLE(SYS.RECOMMEND_REFLECTIONS(ARRAY['<jobId>', '<jobId>']))
```

The ``SYS.RECOMMEND_REFLECTIONS`` function's argument must be an array literal.

Parameters

The ``SYS.RECOMMEND_REFLECTIONS`` table function has one parameter: an array that contains the job IDs for the SQL queries that you want to accelerate.

You can list up to 100 job IDs in each `SYS.RECOMMEND_REFLECTIONS` query. You must have permission to view every job you list in the query.

Output

The output for `SYS.RECOMMEND_REFLECTIONS` queries is a table that includes the following columns:

| Column Name | Data Type | Description |

| --- | --- | --- |

| view_sql | varchar | **If the recommendation is for a default raw Reflection:** The value in this field is `Default Raw Reflection`. **If the recommendation is for an aggregation Reflection:** An SQL command that creates a view based on the initial SQL query in the referenced job ID. Dremio uses this view to create an aggregation Reflection when you run the SQL command from the `reflection_sql` column. Before running this command, ensure that you have already created an Arctic catalog or folder named `recommended_view`.

| reflection_sql | varchar | **If the recommendation is for a default raw Reflection:** An SQL command for defining the Reflection on an existing view. **If the recommendation is for an aggregation Reflection:** An SQL command that alters and adds an aggregation Reflection to the view that is created when you run the SQL command in the `view_sql` column. Before running this command, ensure that you have run the corresponding command in the `view_sql` column. Dremio automatically consolidates recommendations for aggregation Reflections when possible to create a single recommendation for similar SQL queries, so the output table may not contain an individual row that corresponds to each job ID listed in the `SYS.RECOMMEND_REFLECTIONS` query.

| job_ids | list | The unique identifier of the jobs that ran the queries for which the recommendation is given.

| reflection_score | double | Dremio's score for the recommended Reflection's quality, on a scale of 0 (worst) to 100 (best). The reflection_score value considers the recommended Reflection's anticipated quality compared to existing Reflections and other recommended Reflections, as well as the likely improvement in query run times if the recommended Reflection is implemented.

| average_improvement_factor | double | The likely average multiplicative rate of improvement for each query if you implement the recommended Reflection. For example, if the average_improvement_factor value is 2.34, implementing the recommended Reflection is likely to speed up each query by 2.34 times, on average.

| average_improvement_ms | double | The likely average improvement, in milliseconds, for each query if you implement the recommended Reflection. For example, if the average_improvement_ms value is 5400, implementing the recommended Reflection is likely to save an average of 5400 milliseconds for each query that uses the Reflection.

Example Query and Output

To request a recommendation, you must have the job ID for at least one SQL query that you want to accelerate. For example, suppose this is the SQL query:

Example SQL Query

```
SELECT Max("p_size")
FROM s3.bucket1.tpch.sf10.parquet.part
```

```
JOIN s3.bucket1.tpch.sf10.parquet.lineitem
ON "p_partkey" = "l_partkey"
GROUP BY "p_brand"
```

Obtaining Recommendations

Retrieve the job ID for the query.

Run the `SYS.RECOMMEND_REFLECTIONS` query with the job ID. In the following example, the job ID is `844c0023-6272-8b16-aef3-aea289acadb1`:

Example `SYS.RECOMMEND_REFLECTIONS` Query

```
SELECT * FROM
TABLE(SYS.RECOMMEND_REFLECTIONS(ARRAY['844c0023-6272-8b16-aef3-aea289acadb1']))
```

The result of the query is returned in this table:

Example Results

view_sql	reflection_sql	job_ids	reflection_score	average_improvement_factor	average_improvement_ms
---	---	---	---	---	---

```
CREATE VIEW
"recommended\_view"."Dataset\_9d74a03b-747a-42a2-a5ca-7f9c6f77b55d" AS SELECT
"part"."P\_BRAND" AS "F0[P\_BRAND]", "part"."P\_SIZE" AS "F1[P\_SIZE]",
"part"."P\_PARTKEY" AS "extra#0", "part"."P\_NAME" AS "extra#1", "part"."P\_MFGR" AS
"extra#2", "part"."P\_TYPE" AS "extra#4", "part"."P\_CONTAINER" AS "extra#6",
"part"."P\_RETAILPRICE" AS "extra#7", "part"."P\_COMMENT" AS "extra#8",
"lineitem"."L\_ORDERKEY" AS "extra#9", "lineitem"."L\_PARTKEY" AS "extra#10",
"lineitem"."L\_SUPPKEY" AS "extra#11", "lineitem"."L\_LINENUMBER" AS "extra#12",
"lineitem"."L\_QUANTITY" AS "extra#13", "lineitem"."L\_EXTENDEDPRICE" AS
"extra#14", "lineitem"."L\_DISCOUNT" AS "extra#15", "lineitem"."L\_TAX" AS
"extra#16", "lineitem"."L\_RETURNFLAG" AS "extra#17", "lineitem"."L\_LINESTATUS" AS
"extra#18", "lineitem"."L\_SHIPDATE" AS "extra#19", "lineitem"."L\_COMMITDATE" AS
"extra#20", "lineitem"."L\_RECEIPTDATE" AS "extra#21", "lineitem"."L\_SHIPINSTRUCT"
AS "extra#22", "lineitem"."L\_SHIPMODE" AS "extra#23", "lineitem"."L\_COMMENT" AS
"extra#24" FROM "s3"."bucket1"."tpch"."sf10"."parquet"."lineitem" INNER JOIN
"s3"."bucket1"."tpch"."sf10"."parquet"."part" ON "part"."P\_PARTKEY" =
"lineitem"."L\_PARTKEY"
ALTER DATASET
"recommended\_view"."Dataset\_9d74a03b-747a-42a2-a5ca-7f9c6f77b55d" CREATE
AGGREGATE Reflection "agg\_0e07a376-7f8e-4c68-b2ce-6f6e819bebe6" USING
DIMENSIONS ("F0[P\_BRAND]") MEASURES ("F1[P\_SIZE]" (MAX)) |
"6j6c34cf-9drf-b07a-5ab7-abea69a66d00", "1a3c67c0-aab0-f9fb-97b4-af374b520100",
"1a3c67c0-db35-3645-9ef1-2a84e4d0ce00" | 50.00 | 10.00 | 7196 |
```

Using the Recommendation

If a recommendation is for an aggregation Reflection:

Create an Arctic catalog or folder named `recommended_view`.

Run the query listed in the `view_sql` column to create a view based on the initial SQL query.

Run the query listed in the `reflection_sql` column to create the recommended Reflection and wait for the Reflection to finish refreshing.

When the Reflection refresh is complete, run the SQL query to observe the acceleration due to the added Reflection.

Limitations

The `SYS.RECOMMEND_REFLECTIONS` table function has the following limitations:

The function cannot provide recommendations for jobs with multiple nested queries that contain joins.

An SQL query can contain only inner joins. Outer joins that are part of a view definition are also supported. Other types of joins are not supported.

An SQL query cannot contain Window functions.

Sending Requests to the Recommendations API

You can use the Recommendations API to submit the job IDs of jobs that ran SQL queries and receive recommendations for Reflections that can accelerate those queries.

Locations of the Reflections Editor

You use the Reflections editor to create, edit, and remove raw and aggregation Reflections.

If you know which table or view you want to create a Reflection from, follow either of these steps to open the Reflections editor:

In the space in which the anchor is located, click the gear icon in the Action field for the anchor. In the sidebar of the Dataset Settings window, select Reflections.

In the space in which the anchor is located, click the name of the anchor. Then, click Reflections in the bar at the top of the screen.

You can also open the Reflections editor from an existing Reflection that is listed in the Reflections page. Follow these steps to open the Reflections editor:

Open the Reflections page.

Click the gear in the sidebar, and then select Reflections in the sidebar of the Settings page.

Click the name of the Reflection. The Acceleration window is opened. The editor appears in this window.

Creating Raw Reflections

You can use the Reflections editor to create two types of raw Reflection:

A default raw Reflection that includes all of the columns of the anchor, but does not sort or horizontally partition on any columns

A raw Reflection that includes all or a subset of the columns of the anchor, and that does one or both of the following things:

Sorts on one or more columns

Horizontally partitions the data according to the values in one or more columns

note

For creating Reflections on views and tables with row-access and column-masking policies, see [Using Reflections on Datasets with Policies](#).

Prerequisites

If you want to accelerate queries on unoptimized data or data in slow storage, create a view that is itself created from a table in a non-columnar format or on slow-scan storage. You can then create your raw Reflection from that view.

If you want to accelerate "needle-in-a-haystack" queries, create a view that includes a predicate to include only the rows that you want to scan. You can then create your raw Reflection from that view.

If you want to accelerate queries that perform expensive transformations, create a view that performs those transformations. You can then create your raw Reflection from that view.

If you want to accelerate queries that perform joins, create a view that performs the joins. You can then create your raw Reflection from that view.

Creating Default Raw Reflections

In the **Basic** view of the Reflections editor, you can create a raw Reflection that includes all of the fields that are in a table or view. Creating a basic raw Reflection ensures that Dremio never runs user queries against the underlying table or view when the raw Reflection is enabled.

Restrictions of the **Basic** View

You cannot select fields to sort or create horizontal partitions on.

The name of the Reflection that you create is restricted to "Raw Reflection".

You can create only one raw Reflection. If you want to create multiple raw Reflections at a time, use the **Advanced** view.

To create a raw Reflection in the **Basic** view of the Reflections editor:

Open the Reflections editor.

See Locations of the Reflections Editor to find out where you can open the editor from.

Click the toggle switch on the left side of the **Raw Reflections** bar.



Click **Save**.

tip

You can also create raw Reflections using [SQL commands](#).

For tips on what to do now after your raw Reflection is created and enabled, see What to Do Next.

Creating Customized Raw Reflections

In the **Advanced** view of the Reflections editor, you can create one or more raw Reflections that include all or a selection of the fields that are in the anchor or supported anchor. You can also choose sort fields and fields for partitioning horizontally.

Dremio recommends that you follow the best practices listed in [Best Practices for Creating Raw and Aggregation Reflections](#) when you create customized raw Reflections.

If you make any of the following changes to a raw Reflection when you are using the **Advanced** view, you cannot switch to the **Basic** view:

Deselect one or more fields in the **Display** column. By default, all of the fields are selected.

Select one or more fields in the **Sort**, **Partition**, or **Distribute** column.

To create a raw Reflection in the **Advanced** view of the Reflections editor:

Open the Reflections editor.

See Locations of the Reflections Editor to find out where you can open the editor from.

If the **Advanced** view is not already displayed, click the **Advanced View** button in the top-right corner of the editor.

Click the toggle switch in the table labeled **Raw Reflection** to enable the raw Reflection.

Queries do not start using the Reflection, however, until after you finished editing the Reflection and click **Save** in a later step.



(Optional) Click in the label to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note


Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Display** column to include fields in or exclude them from your Reflection.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in Best Practices for Creating Raw and Aggregation Reflections.

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, recommended partition columns and transforms are selected for you. If you change the selection of columns, then this icon appears at the top of the table: . You can click it to revert back to the recommended selection of partition columns.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

- a. Click the gear icon in the table in which you are defining the Reflection.
- b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

What to Do Next

After you create a raw Reflection that is enabled, test whether the query optimizer is making queries use it. See [Testing Reflections](#) for the steps.

When you are sure that the Reflection is being used, follow one of these steps:

If the base table is in the Apache Iceberg format or the Delta Lake format: Set the schedule according to which all Reflections on the table are refreshed.

For all other base tables: Set the refresh type for all Reflections on the underlying table and set the schedule according to which they are refreshed.

See [Refreshing Reflections](#).

Creating Aggregation Reflections

Aggregation Reflections are summarized representations of data. Most BI tools generate aggregation and GROUP BY queries. Aggregation Reflections optimize these kinds of query patterns.

When you create aggregation Reflections, keep in mind these best practices:

Use dimensions that have relatively low cardinality in a table or view. The higher the cardinality of a dimension, the less benefit an aggregation Reflection has on query performance. Lower cardinality aggregation Reflections require less time to scan.

For a single table or view, create one aggregation Reflection for each important subset of dimensions in your queries, rather than one aggregation Reflection that includes all dimensions. Multiple small aggregation Reflections (versus one large one) are good for isolated pockets of query patterns on the same table or view that do not overlap. If your query patterns overlap, use fewer larger aggregation Reflections.

There are two cautions that accompany this advice, however:

Be careful of creating aggregation Reflections that have too few dimensions for your queries.

If a query uses more dimensions than are included in an aggregation Reflection, the Reflection cannot satisfy the query and the query optimizer does not run the query against it.

Be careful of creating more aggregation Reflections than are necessary to satisfy queries against a table or view.

The more Reflections you create, the more time the query optimizer requires to plan the execution of queries. Therefore, creating more aggregation Reflections than you need can slow down query performance, even if your aggregation Reflections are low-cardinality.

If you want to include a computed measure, first create a view with the computed column to use as a measure, and then create the aggregation Reflection on the view.

Dremio recommends that you also follow the best practices listed in [Best Practices for Creating Raw and Aggregation Reflections](#) when you create customized aggregation Reflections.

Creating Default Aggregation Reflections

You can use the **Basic** view of the Reflections editor to create one aggregation Reflection that includes fields, from the anchor or supported anchor, that are recommended for use as dimensions or measures. You can add or remove dimensions and measures, too.

Restrictions

You can create only one aggregation Reflection in the **Basic** view. If you want to create multiple aggregations Reflections at a time, use the **Advanced** view.

You cannot select fields for sorting or horizontally partitioning.

The name of the Reflection is restricted to "Aggregation Reflection".

Procedure

To create an aggregation Reflection in the **Basic** view of the Reflections editor:

Open the Reflections editor.

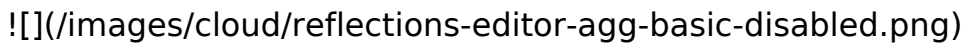
See Locations of the Reflections Editor to find out where you can open the editor from.

In the **Aggregations Reflections** section of the editor, click **Generate** to get recommended fields to use as dimensions and measures. This will override any previously selected dimensions and measures. If you wish to proceed, click **Continue** in the confirmation dialog that follows.



In the **Aggregation Reflection** section of the editor, modify or accept the recommended fields for dimensions and measures.

To make the Reflection available to the query optimizer after you create it, click the toggle switch on the left side of the **Aggregation Reflections** bar.



Click **Save**.

tip

You can also create aggregation Reflections using [SQL commands](#).

For tips on what to do now after your aggregation Reflection is created and enabled, see [What to Do Next](#).

Creating Customized Aggregation Reflections

You can use the **Advanced** view of the Reflections editor to create one or more aggregation Reflections that select which fields in the anchor or supporting anchor to use as dimensions and measures. For each field that you use as a measure, you can use one or more of these SQL functions: APPROX_DISTINCT_COUNT, COUNT, MAX, and MIN. You can also choose sort fields and fields for partitioning horizontally.

Procedure

To create an aggregation Reflection in the **Advanced** view of the Reflections editor:

Open the Reflections editor.

See Locations of the Reflections Editor to find out where you can open the editor from.

Click the **Advanced View** button in the top-right corner of the editor.

Click **Aggregation Reflections**.

The Aggregation Reflections section is displayed, and one table for refining the aggregation Reflection that appeared in the **Basic** view is ready.

(Optional) Click in the name to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Dimensions** column to include or exclude fields to use as dimensions.

Click in the Measures field to include or exclude fields to use as measures. You can use one or more of these SQL functions for each measure: APPROX_DISTINCT_COUNT, COUNT, MAX, and MIN.

The full list of SQL aggregation functions that Dremio supports is not supported in the Reflections editor. If you want to create a Reflection that aggregates data by using the SQL function AVG, CORR, HLL, SUM, VAR_POP, or VAR_SAMP, you must create a view that uses the function, and then create a raw Reflection from that view.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in Best Practices for Creating Raw and Aggregation Reflections.

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, recommended partition columns and transforms are selected for you. If you change the selection of columns, then this icon appears at the top of the table: [!This is the Recommendations icon.](#) You can click it to revert back to the recommended selection of partition columns.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

- a. Click the gear icon in the table in which you are defining the Reflection.
- b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

What to Do Next

After you create an aggregation Reflection that is enabled, test whether the query optimizer is making queries use it. See [Testing Reflections](#) for the steps.

When you are sure that the Reflection is being used, follow one of these steps:

If the base table is in the Apache Iceberg format or the Delta Lake format: Set the schedule according to which all Reflections on the table are refreshed.

For all other base tables: Set the refresh type for all Reflections on the underlying table and set the schedule according to which they are refreshed.

See [Refreshing Reflections](#).

Editing Raw Reflections

You can edit an existing raw Reflection. You might want to do so if you are iteratively designing and testing a raw Reflection, if the definition of the view that the Reflection was created from was changed, or if the schema of the underlying table was changed.

If you created a raw Reflection in the **Basic** view of the Reflections editor, you must use the **Advanced** view to edit it.

Dremio runs the job or jobs to recreate the Reflection after you click **Save**.

Procedure

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

Click the **Advanced View** button in the top-right corner of the editor.

In the **Raw Reflections** section of the **Advanced** view, locate the table that shows the definition of your Reflection.

(Optional) Click in the label to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.


Click in the **Display** column to include fields in or exclude them from your Reflection.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in [Best Practices for Creating Raw and Aggregation Reflections](#).

-

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, partition columns and transforms are recommended for you. Hover over the  icon at the top of the table to see the recommendation. Click the icon to accept the recommendation.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

- a. Click the gear icon in the table in which you are defining the Reflection.
- b. In the field **Reflection execution strategy**, select either of these options:

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

Editing Aggregation Reflections

You might want to edit an aggregation Reflection if you are iteratively designing and testing an aggregation Reflection, if the definition of the view that the Reflection was created from was changed, if the schema of the underlying table was changed, or if you want to revise one or more aggregations defined in the Reflection.

If you created an aggregation Reflection in the **Basic** view of the Reflections editor, you can edit that Reflection either in the **Basic** view or in the **Advanced** view.

Dremio runs the job or jobs to recreate the Reflection after you click **Save**.

Editing Aggregation Reflections in the **Basic** View of the Reflections Editor

Open the Reflections editor.

See Locations of the Reflections Editor to find out where you can open the editor from.

In the Aggregation Reflection section of the editor, modify or accept the recommendation for dimension fields and measure fields.

Click **Save**.

Editing Aggregation Reflections in the **Advanced** View of

the Reflections Editor

Open the Reflections editor.

See [Locations of the Reflections Editor](#) to find out where you can open the editor from.

Click the **Advanced View** button in the top-right corner of the editor.

Click **Aggregation Reflections**.

(Optional) Click in the name to rename the Reflection.

The purpose of the name is to help you understand, when you read job reports, which Reflections the query optimizer considered and chose when planning queries.

In the columns of the table, follow these steps, which you don't have to do in any particular order:

note

Ignore the **Distribution** column. Selecting fields in it has no effect on the Reflection.

Click in the **Dimensions** column to include or exclude fields to use as dimensions.


Click in the Measures field to include or exclude fields to use as measures. You can use one or more of these SQL functions for each measure: APPROX_DISTINCT_COUNT, COUNT, MAX, and MIN.

The full list of SQL aggregation functions that Dremio supports is not supported in the Reflections editor. If you want to create a Reflection that aggregates data by using the SQL function AVG, CORR, HLL, SUM, VAR_POP, or VAR_SAMP, you must create a view that uses the function, and then create a raw Reflection from that view.

Click in the **Sort** column to select fields on which to sort the data in the Reflection. For guidance in selecting a field on which to sort, see the section [Sort Reflections on High-Cardinality Fields](#) in Best Practices for Creating Raw and Aggregation Reflections.

Click in the **Partition** column to select fields on which to horizontally partition the rows in the Reflection. For guidance in selecting fields on which to partition, and which partition transforms to apply to those fields, see the section [Horizontally Partition Reflections that Have Many Rows](#).

note

If the Reflection is based on an Iceberg table, a filesystem source, an AWS Glue source, or a Hive source, and that table is partitioned, partition columns and transforms are recommended for you. Hover over the  icon at the top of the table to see the recommendation. Click the icon to accept the recommendation.

(Optional) Optimize the number of files used to store the Reflection. You can optimize for fast refreshes or for fast read performance by queries. Follow these steps:

a. Click the gear icon in the table in which you are defining the Reflection.

b. In the field **Reflection execution strategy**, select either of these options:

-

Select **Minimize Time Needed To Refresh** if you need the Reflection to be created as fast as possible. This option can result in the data for the Reflection being stored in many small files. This is the default option.

Select **Minimize Number Of Files** when you want to improve read performance of queries against the Reflection. With this option, there tend to be fewer seeks performed for a given query.

Click **Save** when you are finished.

Creating External Reflections

See [External Reflections](#) for a description of what external Reflections are and their benefits.

To create an external Reflection:

Follow these steps in the data source:

- a. Select your source table.
- b. Create a table that is derived from the source table, such as an aggregation table, if you do not already have one.

Follow these steps in Dremio:

- a. Define a view on the derived table in the data source. The definition must match that of the derived table.
- b. Define a new external Reflection that maps the view to the derived table.

note

The data types and column names in the external Reflection must match those in the view that the external Reflection is mapped to.

Example

Suppose you have a data source named ``mySource`` that is connected to Dremio. In that data source, there are (among all of your other tables) these two tables:

``sales``, which is a very large table of sales data.

``sales_by_region``, which aggregates by region the data that is in ``sales``.

You want to make the data in ``sales_by_region`` available to data analysts who use Dremio. However, because you already have the ``sales_by_region`` table created, you do not see the need to create a Dremio table from ``sales``, then create a Dremio view that duplicates ``sales_by_region``, and finally create a Reflection on the view. You would like instead to make ``sales_by_region`` available to queries run from Bi tools through Dremio.

To do that, you follow these steps:

Create a view in Dremio that has the same definition as ``sales_by_region``. Notice that the ``FROM`` clause points to the ``sales`` table that is in your data source, not to a

Dremio table.

Example View

```
CREATE VIEW "myWorkspace"."sales_by_region" AS
SELECT
    AVG(sales_amount) average_sales,
    SUM(sales_amount) total_sales,
    COUNT(*) sales_count,
    region
FROM mySource.sales
GROUP BY region
```

Create an external Reflection that maps the view above to `sales_by_region` in `mySource`.

Example External Reflection

```
ALTER DATASET "myWorkspace"."sales_by_region"
CREATE EXTERNAL Reflection "external_sales_by_region"
USING "mySource"."sales_by_region"
```

The external Reflection lets Dremio's query planner know that there is a table in `mySource` that matches the Dremio view `myWorkspace.sales_by_region` and that can be used to satisfy queries against the view. When Dremio users query `myWorkspace.sales_by_region`, Dremio routes the query to the data source `mySource`, which runs the query against `mySource.sales_by_region`.

Editing External Reflections

If you have modified the DDL of a derived table in your data source, follow these steps in Dremio to update the corresponding external Reflection:

Replace the view with one that has a definition that matches the definition of the derived table. When you do so, the external Reflection is dropped.

Define a new external Reflection that maps the view to the derived table.

Viewing Whether Queries Used Reflections

You can view the list of jobs on the Jobs page to find out whether queries were accelerated by Reflections. The Jobs page lists the jobs that ran queries, both queries from your data consumers and queries run within the Dremio user interface.

To find whether a query used a Reflection:

Find the job that ran the query by looking below the details in each row.

Look for the indicator that one or more Reflections were used. A lightning-bolt icon appears next to the job to indicate that a query was accelerated.

View the job summary by clicking the row that represents the job that ran the query.

The job summary appears in the pane to the right of the list of jobs.

Relationship between Reflections and Jobs

The relationship between a job and a Reflection can be one of the following types:

CONSIDERED: the Reflection is defined on a dataset that is used in the query but was determined to not cover the query (for example the Reflection did not have a field that is used by the query).

MATCHED: a Reflection could have been used to accelerate the query but Dremio determined that it would not provide any benefits or another Reflection was determined to be a better choice.

CHOSEN: a Reflection is used to accelerate the query. Note that multiple Reflections can be used to accelerate queries.

Testing Reflections

You can test whether Reflections that you created are used to satisfy a query without actually running the query. This practice can be helpful when the tables are very large and you want to avoid processing large queries unnecessarily.

To test whether one or more Reflections are used by a query:

In the side navigation bar, click the SQL Runner icon.

In the SQL editor, type ``EXPLAIN PLAN FOR`` and then type or paste in your query.

Click the **Run** button.

When the query has finished, click the **Run** link found directly above the query results to view the job details. Any Reflections used will be shown on the page.

Setting the Expiration Policy for Reflections

Rather than delete a Reflection manually, you can specify how long you want Dremio to retain the Reflection before deleting it automatically.

note

Dremio does not allow expiration policies to be set on external Reflections or Reflections that automatically refresh when Iceberg data changes according to the [refresh policy](#).

To set the expiration policy for all Reflections derived from tables in a data source:

Right-click a data lake or external source.

Select **Edit Details**.

In the sidebar of the Edit Source window, click **Reflection Refresh**.

After making your changes, click **Save**. The changes take effect on the next refresh.

To set the expiration policy on Reflections derived from a particular table:

note

The table must be based on more than one file.

Locate a table.

Click the gear icon to its right.

In the sidebar of the Dataset Settings window, click **Reflection Refresh**.

After making your changes, click **Save**. The changes take effect on the next refresh.

Removing Reflections

You can choose to disable or delete Reflections.

Disabling Reflections

Disabled Reflections become unavailable for use by queries and will not be refreshed manually or according to their schedule.

note

Dremio does not disable external Reflections.

To disable a Reflection:

Open the Reflections editor for the Reflection's anchor or supporting anchor. See [Locations of the Reflections Editor](#).

Follow one of these steps:

If there is only one raw Reflection for the table or view, in the **Basic** view click the toggle switch in the **Raw Reflections** bar.

If there are two or more raw Reflections for the table or view, in the **Advanced** view click the toggle switch for the individual raw Reflection that you want to disable.

If there is only one aggregation Reflection for the table or view, in the **Basic** view click the toggle switch in the **Raw Reflections** bar.

If there are two or more aggregation Reflections for the table or view, in the **Advanced** view click the toggle switch for the individual aggregation Reflection that you want to disable.

Click **Save**. The changes take effect immediately.

Deleting Reflections

You can delete Reflections individually, or all of the Reflections on a table or view. When you delete a Reflection, its definition, data, and metadata are entirely deleted.

To delete a single raw or aggregation Reflection:

Open the Reflections editor for the Reflection's anchor or supporting anchor. See [Locations of the Reflections Editor](#).

Open the **Advanced** view, if it is not already open.

If the Reflection is an aggregation Reflection, click **Aggregation Reflections**.

Click the trash can icon for the Reflection that you want to delete.

Click **Save**. The deletion takes effect immediately.

To delete all raw and aggregation Reflections on a table or view:

Open the Reflections editor for the Reflection's anchor or supporting anchor. See [Locations of the Reflections Editor](#).

Click **Remove All Reflections**.

Click **Save**.

To delete an external Reflection, or to delete a raw or aggregation Reflection without using the Reflections editor, run this SQL command:

Delete a Reflection

```
ALTER DATASET <DATASET_PATH> DROP Reflection <REFLECTION_NAME>
```

DATASET_PATH: The path of the view on which the external Reflection is based.

REFLECTION_NAME: The name of the external Reflection.

Additional Resources

Find out more about Reflections by enrolling in the [Data Reflections Deep Dive course in Dremio University](#).

Was this page helpful?

[Previous

Autonomous Reflections](/current/acceleration/autonomous-reflections)[Next

View	Whether	Queries	Used
Reflections](/current/acceleration/manual-reflections/info-about-queries)			

Reflection Recommendations

Sending Requests to the Recommendations API

Locations of the Reflections Editor

Creating Raw Reflections

Prerequisites

Creating Default Raw Reflections

Creating Customized Raw Reflections

What to Do Next

Creating Aggregation Reflections

Creating Default Aggregation Reflections

Creating Customized Aggregation Reflections

What to Do Next

Editing Raw Reflections

Procedure

Editing Aggregation Reflections

Editing Aggregation Reflections in the **Basic** View of the Reflections Editor

Editing Aggregation Reflections in the **Advanced** View of the Reflections Editor

Creating External Reflections

Example

Editing External Reflections

Viewing Whether Queries Used Reflections

Relationship between Reflections and Jobs

Testing Reflections

Setting the Expiration Policy for Reflections

Removing Reflections

Disabling Reflections

Deleting Reflections

Additional Resources

Source: dremio-software-admin.md

Dremio Software - Admin

Source: <https://docs.dremio.com/current/admin/>

Version: current [26.x]

Administration

This section includes administration topics.

If you're just getting started, we suggest using one of the guides under [Getting Started](#) for one of our platforms.

[Dremio on Kubernetes](#)

[Licensing](#)

[Service Telemetry](#)

[Automated Backup](#)

[Managing Job Workloads](#)

[Monitoring](#)

[Backup and Restore for Open Catalog](#)

[Configure Model Providers](#)

Was this page helpful?

[\[Previous](#)

[Security and Compliance\]](#)(/current/security/)[[Next](#)

[Dremio on Kubernetes\]](#)(/current/admin/admin-dremio-kubernetes/)

Source:
<https://docs.dremio.com/current/admin/admin-dremio-kubernetes/>

Version: current [26.x]

On this page

Administer Dremio on Kubernetes

This section includes topics about administering Dremio on supported Kubernetes environments, including information about monitoring logs, scaling pods, changing configurations, performing basic administrative tasks such as backing up, restoring, cleaning, and upgrading Dremio.

Monitoring Logs and Usage

Monitoring the cluster's resource usage (e.g., heap and direct memory, CPU, disk I/O, etc.) is crucial to maintaining long-term stability as the system scales. For this reason, Dremio recommends setting up a monitoring stack, such as Prometheus and Grafana. For a detailed setup tutorial and an overview of which metrics to track, see [Dremio Monitoring in Kubernetes](#). For more information, see this PDF guide on the [Dremio Enterprise Edition \(Software\) Shared Responsibility Model](#).

Managing Workloads

Most workloads can be handled with a Large (8 executors) or X-Large (12 executors) engine, each with 32 CPUs per executor. Larger engine sizes may be required for certain workloads. Over-parallelization of queries can cause performance degradation. Thus, packing workloads of all shapes or sizes onto a few very large engines is ill-advised. Workloads should be divided into high-cost and low-cost queries, and dedicated queues should be configured for tasks such as Reflections, metadata refresh, and table optimization jobs. These can then be divided between right-sized engines. For more information, see [Dremio's Well-Architected Framework](#).

Changing Your Configuration

If you need to update your configuration, you can do so after the installation by editing the configuration files and then upgrading using an upgrade command, for example:

```
helm upgrade <chart release name> oci://quay.io/dremio/dremio-helm -f  
<your-local-path>/values-overrides.yaml --version <helm-chart-version>
```

The upgrade process pushes your changes to all pods in your Kubernetes cluster and restarts the pods.

For example, to permanently change the resources of your coordinator pod:

Edit the `values-overrides.yaml` file and change the resources specified for the coordinator. In this example, `memory` is `32Gi` and `cpu` is `8`.

```
coordinator:  
  resources:  
    limits:  
      memory: 32Gi  
    requests:  
      cpu: 8  
      memory: 32Gi
```

Run the upgrade command. Replacing the template values:

```
helm upgrade <chart release name> oci://quay.io/dremio/dremio-helm -f  
<your-local-path>/values-overrides.yaml --version <helm-chart-version>
```

note

If the command takes longer than a few minutes to finish, check the status of the pods with the `kubectl get pods` command. If the pods are pending scheduling due to limited memory or CPU, adjust the values you specified for the properties in the `values-overrides.yaml` file or add more resources to your Kubernetes cluster.

Using Support Keys

Use [support keys](#) only when instructed by Dremio Support. If misused, they can alter the application's behavior and lead to unexpected failures.

Using the Dremio Admin CLI on Kubernetes

The [Dremio Admin CLI](#) is the mechanism to back up, restore, add internal users, etc. For more information on the various commands the see CLI reference previously linked. In order to run the CLI commands you need to access either the `dremio-master-0` or `dremio-admin` pod. This requires the use of the `kubectl` command line tool and access to the Kubernetes cluster and namespace where Dremio is deployed.

note

The term `master` is a legacy label used in this command. We now refer to this as the main coordinator pod.

Some CLI commands like [Back Up Dremio](#) require Dremio to be **online**. This means Dremio must be deployed normally per [Deploying Dremio to Kubernetes](#). When inspecting Dremio's pods, `dremio-master-0` must be present and `RUNNING` to be considered **online**.

Some CLI commands like [Clean](#) require Dremio to be **offline**. To use them, Dremio must be deployed and running in admin mode. If not, you must redeploy Dremio in admin mode. The requirements section for each command will note whether Dremio should be online or offline. If it is not mentioned, then the command will work in either case.

To redeploy Dremio in admin mode, you must run a `helm upgrade` command where the `DremioAdmin` flag is set to `true`. Here is a templated example command:

```
helm upgrade <chart-release-name> oci://quay.io/dremio/dremio-helm -f
<your-local-path>/values-overrides.yaml --version <helm-chart-version> --set
DremioAdmin=true
```

This command will cause the shutdown of the Coordinators and Executors. In their place will start the `dremio-admin` pod. Crucially, this pod will mount the `dremio-master-0` volume allowing for operations on the constituent KV store.

To get command line access to the `dremio-master-0`, `dremio-admin`, or any pod for that matter, you would use the `kubectl exec` command. Here is an example using the `-it` option for interactive, and the [bash` option to enter a bash session:](#)

```
kubectl exec -it <pod-name> -- bash
```

Once you've entered the pod, you can run typical shell commands to explore the file system and execute commands. For more information, see [kubectl exec](#).

The `dremio-admin` utility is within the `/opt/dremio/bin` directory of both the main and admin pods and can be used to execute the various [Dremio Admin CLI](#) commands.

To exit Dremio admin mode and restart the normal service, you must redeploy Dremio again using the command above and setting only `DremioAdmin=false`.

Was this page helpful?

[Previous

Administration]/(current/admin)/[Next

Upgrade Dremio](/current/admin/admin-dremio-kubernetes/upgrade)

Monitoring Logs and Usage

Managing Workloads

Changing Your Configuration

Using Support Keys

Using the Dremio Admin CLI on Kubernetes

Source:
<https://docs.dremio.com/current/admin/licensing/>

Version: current [26.x]

On this page

Licensing

As of Dremio 26.0, a new licensing model has been introduced, affecting Dremio deployments on Kubernetes. Understanding these requirements is crucial for ensuring a smooth deployment and uninterrupted operation.

License Key Requirement

A valid license key is mandatory for deploying a Dremio cluster on Kubernetes. Without it, the cluster will fail to start. This requirement applies to both new installations and upgrades.

Additionally, Dremio's telemetry functionality relies on a valid license key. Without one, telemetry data will not be accurately reported, potentially affecting system monitoring and analytics. For more information about telemetry, see [Service Telemetry for Kubernetes Deployments](#).

tip

Do you know you can obtain a **free trial license key** for Dremio? Learn more about it in [Dremio Enterprise Edition Free Trial](#).

Obtain a License Key

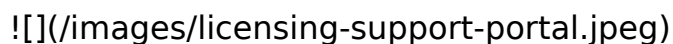
Acquiring a license key for Dremio 26.0+ is a fully automated process where customers generate their own license keys through the [Dremio Support Portal](#), without needing to contact Dremio Support or Sales. The process is self-service:

Go to the [Dremio Support Portal](#) and click `Submit a request`. Create a support ticket and, in the **Subject** field, be sure to include "License Request". This helps us identify and route your request quickly.

After the support ticket is created, you'll receive an automated response message

asking you to confirm whether you are requesting a new or replacement license specifically for deploying Dremio 26.0+ on Kubernetes. Reply to confirm. This step is important to ensure your ticket is routed correctly.

Once you confirm your request, you'll automatically receive the appropriate license file required to deploy your Dremio cluster. No further action is needed unless Dremio Support or Sales follows up with additional questions.



When you download your license file, it should look like this:

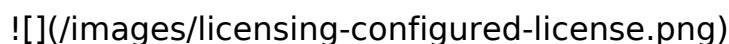


Use Your License Key for Kubernetes Deployment

To ensure your Dremio cluster starts up correctly, follow these steps to apply your license key:

Open the license file you have obtained in the support ticket (see the section above on [How to Obtain a License Key](#)), and copy the license key string.

Go to your Kubernetes deployment configuration file and paste the license key string to the `license:` property enclosed in double quotation marks (`" "`). For more information about the configuration file, see [Configuring Your Values](#).



With the license key set in your configuration file, you can proceed to [Deploying Dremio to Kubernetes](#).

Verify Your License

During startup, Dremio will validate the license key using its internal mechanisms (connectivity to the Internet is not required). If the key is missing, incorrect, expired, or invalid, the cluster will not start, and an error message will be logged to indicate the issue. If the license is valid, the cluster will start as expected.

Troubleshooting Tips

If you encounter any verification issues, do the following:

Double-check that the license key was copied correctly (no extra spaces or line breaks) and is enclosed in double-quotation marks (`" "`).

Review the Dremio logs for license-related error messages.

Reach out via your support ticket for further assistance.

Was this page helpful?

[\[Previous](#)

[Upgrade Dremio\]](#)</current/admin/admin-dremio-kubernetes/upgrade>][\[Next](#)

[Enterprise Edition Free Trial\]\(/current/admin/licensing/free-trial\)](#)

[License Key Requirement](#)

[Obtain a License Key](#)

[Use Your License Key for Kubernetes Deployment](#)

[Verify Your License](#)

[Troubleshooting Tips](#)

Source:
<https://docs.dremio.com/current/admin/billing/>

Version: current [26.x]

On this page

Dremio Billing on Kubernetes

Dremio on Kubernetes uses a consumption-based billing model where costs are calculated based on your actual usage of compute resources, measured in Dremio Consumption Units (DCUs). Billing is managed through Orb, which tracks your DCUs and handles invoicing on a monthly basis (the billing period starts at 00:00:00 on the first day of the month and ends at 23:59:59 on the last day of the month).

Prerequisites

Have an account in Orb. To get access, contact Dremio Support.

Service Telemetry must be enabled to transmit your usage data to Orb.

View Your Billing

To access your billing information:

In your browser, navigate to Orb.

Log in to your Orb account.

Once logged in, you will see a dashboard with your account information, unpaid invoices, and the DCUs for the current billing period. You will also have access to your invoice history, where you can view and download the details of each invoice.

Was this page helpful?

[\[Previous](#)

[Enterprise Edition Free Trial\]\(/current/admin/licensing/free-trial\)\[Next](#)

[Usage\]\(/current/admin/billing/usage\)](#)

[Prerequisites](#)

Source:
<https://docs.dremio.com/current/admin/model-providers/>

Version: current [26.x]

On this page

Configure Model Providers Enterprise

Starting with Dremio Software 26.1, you can configure model providers for AI functionality when deploying Dremio clusters on Kubernetes. After you configure at least one model provider, you must set a default model provider and optionally set an allowlist of available models. Dremio uses this default provider for all Dremio's AI Agent interactions and whereas the allowlist models can be used by anyone writing AI functions.

Supported Model Providers

Dremio supports configuration of the following model providers and models. Dremio recommends using enterprise-grade reasoning models for the best performance and experience.

Category	Models	Connection Method(s)
OpenAI	* gpt-5-2025-08-07 * gpt-5-mini-2025-08-07 * gpt-5-nano-2025-08-07 * gpt-4.1-2025-04-14 * gpt-4o-2024-11-20 * gpt-4-turbo-2024-04-09 * gpt-4.1-mini-2025-04-14 * o3-mini-2025-01-31 * o4-mini-2025-04-16 * o3-2025-04-16	* Access Key
Anthropic	* claude-sonnet-4-5-20250929 * claude-opus-4-1-20250805 * claude-opus-4-20250514 * claude-sonnet-4-20250514	* Access Key
Google Gemini	* gemini-2.5-pro	* Access Key
AWS Bedrock	* specify Model ID(s) * AWS Bedrock Supported Models	* Access Key * IAM Role
Azure OpenAI	* specify Deployment Name(s) * Azure Supported Models	Combination of 1. Resource Name 2. Directory ID 3. Application ID 4. Client Secret Value

Add Model Provider

For steps on adding an AWS Bedrock model provider, see [Configure AWS Bedrock as a Model Provider](#).

For all other model providers, follow these steps to add a model provider in the Dremio console:

Click [!This is the Settings icon.](#) in the side navigation bar to go to the Settings page.

Select **Preferences** in the settings sidebar.

Enable the **AI Features** flag.

Click **Add model provider**.

Default Model Provider

To delete the model provider, you must assign a new default unless you are deleting the last available model provider. To update the default model provider to a new one, you must have MODIFY privilege on both the current default and the new proposed default model provider.

Was this page helpful?

[Previous

Usage](/current/admin/billing/usage)[Next

AWS Bedrock](/current/admin/model-providers/aws-bedrock)

Supported Model Providers

Add Model Provider

Default Model Provider

Source:
<https://docs.dremio.com/current/admin/service-telemetry-kubernetes>

Version: current [26.x]

On this page

Service Telemetry for Kubernetes Deployments Enterprise

As of Dremio 26.0, enterprise customers deploying Dremio clusters on Kubernetes will automatically transmit telemetry data back to Dremio's corporate endpoint. This telemetry provides valuable insights into system performance and health, and is also used in the calculations for usage-based billing. Telemetry can be disabled through configuration; however, this is not considered a best practice because telemetry helps Dremio ensure stability and timely support.

Telemetry Data Collection

Dremio's telemetry data collection is strictly limited to operational and performance metrics. These metrics provide visibility into various components and services, ensuring optimal performance and reliability.

Importantly, no customer content (e.g., business data) or user-entered information is transmitted. If you would like to develop a deeper understanding of the metrics transmitted and their contents, you can set up your own internal monitoring of your Dremio cluster by following the steps in Monitoring Dremio Nodes.

The collected telemetry data is categorized as follows:

Category	Description

| **Application Metrics** | These metrics provide insights into the usage and performance of objects within a Dremio deployment, including: * Number of queries, Reflections, sources, and views. * Success and failure rates of queries. * Success and failure rates of Reflection and source refresh operations. |

| **Java Metrics** | These metrics capture internal Java Virtual Machine (JVM) performance indicators from containers running the Dremio application, such as: * Number of active threads. * Memory allocation and usage. * Garbage collection activity and pauses. |

| **Service Metrics** | These metrics measure the health of core components supporting Dremio's execution and coordination services, including: * KVstore performance. * Zookeeper availability and network health. |

| **Kubernetes Metrics** | These metrics provide insight into container and pod behavior for all containers in a Dremio deployment, including: * CPU, memory, and storage requests. * Container restarts and readiness. * StatefulSet desired and current pod count. |

Telemetry Transmission Requirements

Telemetry transmission to Dremio follows the [Dremio Subscription Agreement](#).

Network Requirements for Telemetry Transmission

To ensure successful telemetry transmission, the following network configurations must be in place:

Your network must allow traffic egress to Dremio's endpoint ``observability.dremio.com``.

Dremio's OpenTelemetry collectors use port 443 for secure data transmission via TLS.

Setting Up a Proxy

If traffic egresses to the endpoint and the port is restricted, a proxy can be configured to enable telemetry transmission:

Edit your ``yaml`` configuration file to deploy Dremio to Kubernetes. For more information, refer to [Configuring Your Values](#).

Add your proxy configuration values to the ``yaml`` file using the following syntax:

HTTPS Proxy (Recommended)

HTTP Proxy

```
telemetry:
  extraEnvs: []
  - name: HTTPS_PROXY
    value: https://proxy.example.com:443
```

```
telemetry:
```

```
extraEnvs: []
  - name: HTTP_PROXY
    value: http://proxy.example.com:3128
```

Troubleshooting or Support

For troubleshooting or support, please contact your account representative or Dremio Support.

Related Topics

[Licensing](#) - Learn more about Dremio's licensing and telemetry.

[Billing](#) - Learn more about Dremio's billing and usage data.

Was this page helpful?

[Previous

AWS Bedrock](/current/admin/model-providers/aws-bedrock)[Next

Automated Backup](/current/admin/automated-backups)

Telemetry Data Collection

Telemetry Transmission Requirements

Network Requirements for Telemetry Transmission

Setting Up a Proxy

Troubleshooting or Support

Related Topics

Source:
<https://docs.dremio.com/current/admin/automated-backups>

Version: current [26.x]

Automated Backup Enterprise

Backups are crucial to restoring Dremio's state in case of a critical failure, so we recommend creating regular backups to minimize loss from the restore point. Use Dremio's automated backups to create backups on a schedule and output the backups to a folder in distributed storage.

To enable automated backups, use the ``dremio.automated_backups.enabled`` [support key](#). By default, when ``dremio.automated_backups.enabled`` is enabled, Dremio creates a backup at midnight local time. To set your backup for a different time, set the ``dremio.automated_backups.schedule`` support key to the desired time in ISO format.

For example, `13:45+02:00` or `14:12Z`.

note

Automated backups currently do not create backups for Open Catalog's metadata storage backend. Follow the instructions on Open Catalog backup and restore processes to back up the metadata storage backend for Open Catalog.

Was this page helpful?

[Previous

Service Telemetry](/current/admin/service-telemetry-kubernetes)[Next

Managing Job Workloads](/current/admin/workloads/)

Source:
<https://docs.dremio.com/current/admin/workloads/>

Version: current [26.x]

Managing Job Workloads

Dremio job workloads are managed differently depending on which edition (Enterprise or Community) you are using.

Job History & Job Details

The results of each query, or job, is displayed here in table format. Individual job details may be viewed by clicking on each row.

Jobs may be sorted and arranged by data type based on the columns displaying. Additional information may also be seen for individual jobs by hovering over a row and column for condensed results. By clicking on the desired job, this launches the Job Details screen, which is rich with easy-to-read data regarding how a query was executed from start to finish.

Queue Control

This feature provides basic capability to manage query queues, Reflection queues, query memory, and query thresholds.

Workload Management Enterprise

This feature provides advanced capability to manage cluster resources and workloads by defining a queue with specific characteristics (such as memory limits, CPU priority, and queueing and runtime timeouts) and then defining rules that specify which query is assigned to which queue.

Was this page helpful?

[Previous

Monitoring](/current/admin/monitoring/)[Next

Queue Control](/current/admin/workloads/job-queues)

Version: current [26.x]

On this page

Monitoring

As an administrator, you can monitor logs, usage, system telemetry, [jobs](#), and [Dremio nodes](#).

As the [Dremio Shared Responsibility Models](#) outline, monitoring is a shared responsibility between Dremio and you. The Shared Responsibility Models lay out Dremio's responsibilities for providing monitoring technologies and logs and your responsibilities for implementation and use.

Logs

Logs are primarily for troubleshooting issues and monitoring the health of the deployment.

note

By default, Dremio uses the following locations to write logs:

Tarball - ``<DREMIO_HOME>/log``

RPM - ``/var/log/dremio``

Kubernetes - ``/opt/dremio/log``

Log Types

Log Type	Description
----------	-------------

---	---
-----	-----

Audit	The <code>`audit.json`</code> file tracks all activities that users perform within Dremio. For details, see Audit Logging .
-------	---

System	The following system logs are enabled by default: <ul style="list-style-type: none">* <code>`access.log`</code>: HTTP access log for the Dremio web server; generated by coordinator nodes only.* <code>`server.gc`</code>: Garbage collection log.* <code>`server.log`</code> and <code>`json/server.json`</code>: Server logs generated in a text format (<code>server.log</code>) and json format (<code>json/server.json</code>). Users granted the <code>`ADMIN`</code> role can disable one of these formats.* <code>`server.out`</code>: Log for Dremio daemon standard out.* <code>`metadata_refresh.log`</code>: Log for refreshing metadata.* <code>`tracker.json`</code>: Tracker log.* <code>`vacuum.json`</code>: Log for the files scanned and deleted by <code>`VACUUM CATALOG`</code> and <code>`VACUUM TABLE`</code> commands.
--------	---

Query	Query logging is enabled by default. The <code>`queries.json`</code> file contains the log of completed queries; it does not include queries currently in planning or execution. You can retrieve the same information that is in <code>`queries.json`</code> using the <code>`sys.jobs_recent`</code> system table. Query logs include the following information: <ul style="list-style-type: none">* <code>`queryId`</code>: Unique ID of the executed query.* <code>`queryText`</code>: SQL query text.* <code>`start`</code>: Start time of the query.* <code>`finish`</code>: End time of the query.* <code>`outcome`</code>: Whether the query was completed or
-------	---

failed. * `username`: User who executed the query. * `commandDescription`: Type of the command; may be a regular SQL query execution job or another SQL command. The query log may contain additional information depending on your Dremio configuration. |

| Warning | The `hive.deprecated.function.warning.log` file contains warnings for Hive functions that have been deprecated. To resolve warnings that are listed in this file, replace deprecated functions with a [supported function](#). For example, to resolve a warning that mentions `NVL`, replace `NVL` with `COALESCE`. |

Retrieving Logs from the Dremio Console Enterprise

Retrieve logs for Kubernetes deployments in the Dremio console at **Settings > Support > Download Logs**.

Prerequisites

You must be using Dremio 25.1+. Log collection is powered by Dremio Diagnostics Collector (DDC).

You must have the EXPORT DIAGNOSTICS privilege to view **Download Logs** options in **Settings > Support**.

Downloading Logs

To download logs:

In the Dremio console, navigate to **Settings > Support > Download Logs** and click **Start collecting data**.

note

You may store a maximum of three log bundles. Delete log bundles as needed to start a new log collection if you reach the maximum.

We recommend the default `Light` collection, which provides 7 days of logs and completed queries in the `queries.json` file, for troubleshooting most issues. For more complex issues, select the `Standard` collection, which provides 7 days of logs and 28 days of completed queries in the `queries.json` file.

When Dremio completes log collection, the log bundle appears in a list below **Start collecting data**. To download a log bundle, click **Download** next to the applicable bundle. Log bundles are available to download for 24 hours.

Logging in Kubernetes

By default, all logs are written to a persisted volume mounted at `/opt/dremio/log`.

To disable logging, set `writeLogsToFile: false` in the `values-overrides.yaml` configuration file either globally or individually for each `coordinator` and `executor` parent. For more information, see [Configuring Your Values](#).

Using the Container Console

All logs are written to the container's console (stdout) simultaneously. These logs can be monitored using a `kubectl` command:

Command for viewing logs using kubectl logs

```
kubectl logs [-f] [container-name]
```

Use the `-f` flag to continuously print new log entries to your terminal as they are generated.

You can also write logs to a file on disk in addition to stdout. Read [Writing Logs to a File](#) for details.

Using the AKS Container

Azure provides integration with AKS clusters and Azure Log Analytics to monitor container logs. This is a standard practice that puts infrastructure in place to aggregate logs from containers into a central log store to analyze them.

AKS log monitoring is useful for the following reasons:

Monitoring logs across lots of pods can be overwhelming.

When a pod (for example, a Dremio executor) crashes and restarts, only the logs from the last pod are available.

If a pod is crashing regularly, the logs are lost, which makes it difficult to analyze the reasons for the crash.

For more information regarding AKS, see [Azure Monitor features for Kubernetes monitoring](#).

Enabling Log Monitoring

You can enable log monitoring when creating an AKS cluster or after the cluster has been created.

Once logging is enabled, all your container `stdout` and `stderr` logs are collected by the infrastructure for you to analyze.

While creating an AKS cluster, enable container monitoring. You can use an existing Log Analytics workspace or create a new one.

In an existing AKS cluster where monitoring was not enabled during creation, go to **Logs on the AKS cluster** and enable it.

Viewing Container Logs

To view all the container logs:

Go to **Monitoring > Logs**.

Use the filter option to see the logs from the containers that you are interested in.

Usage

Monitoring usage across your cluster makes it easier to observe patterns, analyze the resources being consumed by your data platform, and understand the impact on your users.

Catalog Usage Enterprise

Go to **Settings** > **Monitor** to view your catalog usage. You must be a member of the `ADMIN` role to access the Monitor page. When you open the Monitor page, you are directed to the Catalog Usage tab by default where you can see the following metrics:

Top 10 most queried datasets and how often the jobs on the dataset were accelerated

Top 10 most queried spaces and source folders

note

A source can be listed in the top 10 most queried spaces and source folders if the source contains a child dataset that was used in the query (for example, `postgres.accounts`). Queries of datasets in sub-folders (for example, `s3.mybucket.iceberg_table`) are classified by the sub-folder and not the source.

All datasets are assessed in the metrics on the Monitor page except for datasets in the system tables, the information schema, and home spaces.

The metrics on the Monitor page analyze only user queries. Refreshes of data Reflections and metadata refreshes are excluded.

Jobs Enterprise

Go to **Settings** > **Monitor** > **Jobs** to open the Jobs tab. You must be a member of the `ADMIN` role to access the Monitor page. The Jobs tab shows an aggregate view of the following metrics for the jobs that are running on your cluster:

Total job count over the last 24 hours and the relative rate of failure/cancelation

Top 10 most active users based on the number of jobs they ran

Total jobs accelerated, total job time saved, and average job speedup from Autonomous Reflections over the past month.

Total number of jobs accelerated by autonomous and manual Reflections over time

Total number of completed and failed jobs over time

Jobs (completed and failed) grouped by the queue they ran on

Percentage of time that jobs spent in each state

Top 10 longest running jobs

To view all jobs and the details of specific jobs, see Viewing Jobs.

Resources Enterprise

Go to **Settings** > **Monitor** > **Resources** to open the Resources tab. You must be a member of the `ADMIN` role to access the Monitor page. The Resources tab shows an aggregate view of the following metrics for the jobs and nodes running on your cluster:

Percentage of CPU and memory utilization for each coordinator and executor node

Top 10 most CPU and memory intensive jobs

Number of running executors

Cluster Usage

Dremio displays the number of unique users who executed jobs on that day and the number of executed jobs.

Hover over  [icon represents help](#) in the side navigation bar.

Click **About Dremio** in the menu.

Click the **Cluster Usage Data** tab.

 "Viewing Cluster Usage")

System Telemetry

Dremio exposes system telemetry metrics in Prometheus format by default. It is not necessary to configure an exporter to collect the metrics. Instead, you can specify the host and port number where metrics are exposed in the [dremio.conf](#) file and scrape the metrics with any Prometheus-compliant tool.

To specify the host and port number where metrics are exposed, add these two properties to the `dremio.conf` file:

`services.web-admin.host`: set to the desired host address (typically `0.0.0.0` or the IP address of the host where Dremio is running).

`services.web-admin.port`: set to any desired value that is greater than `1024`.

For example:

Example host and port settings in dremio.conf

```
services.web-admin.host: "127.0.0.1"
services.web-admin.port: 9090
```

Restart Dremio after you update the `dremio.conf` file to make sure your changes take effect.

Access the exported Dremio system telemetry metrics at `http://<yourHost>:<yourPort>/metrics`.`

For more information about Prometheus metrics, read [Types of Metrics](#) in the Prometheus documentation.

Was this page helpful?

[Previous

Workload Management](/current/admin/workloads/workload-management)[Next

Viewing Jobs](/current/admin/monitoring/jobs/)

Logs

Log Types

Retrieving Logs from the Dremio Console Enterprise

Logging in Kubernetes

Usage

Catalog Usage Enterprise

Jobs Enterprise

Resources Enterprise

Cluster Usage

System Telemetry

Source:
<https://docs.dremio.com/current/admin/open-catalog-backing-store-backuprestore>

Version: current [26.x]

On this page

Open Catalog Backup and Restore Enterprise

Regular backups are essential for protecting Open Catalog metadata and ensuring business continuity. This section explains how to back up and restore the MongoDB cluster that stores Open Catalog's configuration, table metadata, and access control policies.

note

Ensure you have enabled [automated backup](#) for your Dremio cluster before backing up the Open Catalog.

Automated Backups

Automated MongoDB backup is enabled in your ``values-overrides.yaml``. The backups are automatically written to your distributed storage and must be taken while Dremio is operational. Not all object store authentication methods are supported by this feature. See [Configuring the Distributed Storage](#) for details on supported configurations.

When enabled, a backup agent will be deployed into the cluster as a container of the first MongoDB pod ``dremio-mongodb-rs0-0``. Inspect the agent logs with the command:

``kubectl logs -f dremio-mongodb-rs0-0 -c backup-agent -n <your-namespace>``. Backups are written to the ``catalog-backups`` folder of Dremio's distributed storage. The backup names will follow a consistent pattern, for example, ``cron-dremio-mongodb-20251112124000-87jl7``.

Restore

Prerequisites

Ensure that Dremio is in **Admin Mode**. See [Using the Dremio Admin CLI on Kubernetes](#) to understand how to switch to **Admin Mode**.

Export your Kubernetes namespace as an environment variable. Replace the ``<namespace>`` placeholder with your value:

```
export NAMESPACE = <namespace>
```

Run the following command for a list of available backups for the restore:

```
kubectl get psmdb-backup -n $NAMESPACE
```

Run the following command for MongoDB cluster information. The ``clustername`` will be required to start the restore.

```
kubectl get psmdb -n $NAMESPACE
```

Restore From a Full Backup

Restore based on the name of the specific backup.

Create a file named ``restore.yaml``. Fill in the YAML based on the output from the prerequisites, namely: ``<my-cluster-name>`` and ``<my-backup-name>``. Dremio recommends substituting ``<my-restore-name>`` with a name containing the date the restore was performed.

```
apiVersion: psmdb.dremio.com/v1
kind: PerconaServerMongoDBRestore
metadata:
  name: <my-restore-name>
spec:
  clusterName: <my-cluster-name>
  backupName: <my-backup-name>
```

Start the restore by applying the YAML created in the previous step:

```
kubectl apply -f restore.yaml -n $NAMESPACE
```

Once completed, bring Dremio back online. See [Using the Dremio Admin CLI on](#)

[Kubernetes](#) to understand how to leave **Admin Mode**.

Point-in-time Recovery

Restore to a particular point in time within a given backup. This allows for a more granular restore.

Use this command to get a list of all restore times available within a backup.

```
kubectl get psmdb-backup <backup_name> -n $NAMESPACE -o jsonpath='{.status.latestRestorableTime}'
```

Modify the `restore.yaml` specifying your chosen restore date and time in the following format `YYYY-MM-DD HH:MM:SS` from those available.

```
apiVersion: psmdb.dremio.com/v1
kind: PerconaServerMongoDBRestore
metadata:
  name: <my-restore-name>
spec:
  clusterName: <my-cluster-name>
  backupName: <my-backup-name>
  pitr:
    type: date
    date: YYYY-MM-DD hh:mm:ss
```

Start the restore by applying the YAML created in the previous step:

```
kubectl apply -f restore.yaml -n $NAMESPACE
```

Once completed, bring Dremio back online. See [Using the Dremio Admin CLI on Kubernetes](#) to understand how to leave **Admin Mode**.

Was this page helpful?

[Previous

Splunk](/current/admin/monitoring/exporting/splunk)[Next

Developer Guide](/current/developer/)

Automated Backups

Restore

Prerequisites

Restore From a Full Backup

Point-in-time Recovery

Source:

Version: current [26.x]

On this page

Service Telemetry for Kubernetes Deployments Enterprise

As of Dremio 26.0, enterprise customers deploying Dremio clusters on Kubernetes will automatically transmit telemetry data back to Dremio's corporate endpoint. This telemetry provides valuable insights into system performance and health, and is also used in the calculations for usage-based [billing](#). Telemetry can be disabled through configuration; however, this is not considered a best practice because telemetry helps Dremio ensure stability and timely support.

Telemetry Data Collection

Dremio's telemetry data collection is strictly limited to operational and performance metrics. These metrics provide visibility into various components and services, ensuring optimal performance and reliability.

Importantly, no customer content (e.g., business data) or user-entered information is transmitted. If you would like to develop a deeper understanding of the metrics transmitted and their contents, you can set up your own internal monitoring of your Dremio cluster by following the steps in [Monitoring Dremio Nodes](#).

The collected telemetry data is categorized as follows:

| Category | Description |

| --- | --- |

| **Application Metrics** | These metrics provide insights into the usage and performance of objects within a Dremio deployment, including: * Number of queries, Reflections, sources, and views. * Success and failure rates of queries. * Success and failure rates of Reflection and source refresh operations. |

| **Java Metrics** | These metrics capture internal Java Virtual Machine (JVM) performance indicators from containers running the Dremio application, such as: * Number of active threads. * Memory allocation and usage. * Garbage collection activity and pauses. |

| **Service Metrics** | These metrics measure the health of core components supporting Dremio's execution and coordination services, including: * KVstore performance. * Zookeeper availability and network health. |

| **Kubernetes Metrics** | These metrics provide insight into container and pod behavior for all containers in a Dremio deployment, including: * CPU, memory, and storage requests. * Container restarts and readiness. * StatefulSet desired and current pod count. |

Telemetry Transmission Requirements

Telemetry transmission to Dremio follows the [Dremio Subscription Agreement](#).

Network Requirements for Telemetry Transmission

To ensure successful telemetry transmission, the following network configurations must be in place:

Your network must allow traffic egress to Dremio's endpoint ``observability.dremio.com``.

Dremio's OpenTelemetry collectors use port 443 for secure data transmission via TLS.

Setting Up a Proxy

If traffic egresses to the endpoint and the port is restricted, a proxy can be configured to enable telemetry transmission:

Edit your ``.yaml`` configuration file to deploy Dremio to Kubernetes. For more information, refer to [Configuring Your Values](#).

Add your proxy configuration values to the ``.yaml`` file using the following syntax:

HTTPS Proxy (Recommended)

HTTP Proxy

```
telemetry:
  extraEnvs: []
    - name: HTTPS_PROXY
      value: https://proxy.example.com:443
```

```
telemetry:
  extraEnvs: []
    - name: HTTP_PROXY
      value: http://proxy.example.com:3128
```

Troubleshooting or Support

For troubleshooting or support, please contact your account representative or Dremio Support.

Related Topics

[Licensing](#) - Learn more about Dremio's licensing and telemetry.

[Billing](#) - Learn more about Dremio's billing and usage data.

Was this page helpful?

[Previous

AWS Bedrock](/current/admin/model-providers/aws-bedrock)[Next

Automated Backup](/current/admin/automated-backups)

Telemetry Data Collection

- Telemetry Transmission Requirements
- Network Requirements for Telemetry Transmission
- Setting Up a Proxy
- Troubleshooting or Support
- Related Topics

Source:
<https://docs.dremio.com/current/admin/model-providers>

Version: current [26.x]

On this page

Configure Model Providers Enterprise

Starting with Dremio Software 26.1, you can configure model providers for AI functionality when deploying Dremio clusters on Kubernetes. After you configure at least one model provider, you must set a default model provider and optionally set an allowlist of available models. Dremio uses this default provider for all Dremio's AI Agent interactions and whereas the allowlist models can be used by anyone writing AI functions.

Supported Model Providers

Dremio supports configuration of the following model providers and models. Dremio recommends using enterprise-grade reasoning models for the best performance and experience.

Category	Models	Connection Method(s)
---	---	---
OpenAI	* gpt-5-2025-08-07 * gpt-5-mini-2025-08-07 * gpt-5-nano-2025-08-07 * gpt-4.1-2025-04-14 * gpt-4o-2024-11-20 * gpt-4-turbo-2024-04-09 * gpt-4.1-mini-2025-04-14 * o3-mini-2025-01-31 * o4-mini-2025-04-16 * o3-2025-04-16	* Access Key
Anthropic	* claude-sonnet-4-5-20250929 * claude-opus-4-1-20250805 * claude-opus-4-20250514 * claude-sonnet-4-20250514	* Access Key
Google Gemini	* gemini-2.5-pro	* Access Key
AWS Bedrock	* specify Model ID(s) * AWS Bedrock Supported Models	* Access Key * IAM Role
Azure OpenAI	* specify Deployment Name(s) * Azure Supported Models	Combination of 1. Resource Name 2. Directory ID 3. Application ID 4. Client Secret Value

Add Model Provider

For steps on adding an AWS Bedrock model provider, see [Configure AWS Bedrock as a Model Provider](#).

For all other model providers, follow these steps to add a model provider in the Dremio console:

Click [!This is the Settings icon.](#) in the side navigation bar to go to the Settings page.

Select **Preferences** in the settings sidebar.

Enable the **AI Features** flag.

Click **Add model provider**.

Default Model Provider

To delete the model provider, you must assign a new default unless you are deleting the last available model provider. To update the default model provider to a new one, you must have MODIFY privilege on both the current default and the new proposed default model provider.

Was this page helpful?

[\[Previous](#)

[Usage\]\(/current/admin/billing/usage\)\[Next](#)

[AWS Bedrock\]\(/current/admin/model-providers/aws-bedrock\)](#)

[Supported Model Providers](#)

[Add Model Provider](#)

[Default Model Provider](#)

Source: dremio-software-client-applications.md

Dremio Software - Client Applications

Source:
<https://docs.dremio.com/current/client-applications/>

Version: current [26.x]

On this page

Connect Client Applications to Dremio

You can connect to Dremio from a variety of client applications. Connections are established using JDBC or ODBC.

Dremio supports a broad range of clients including the following applications:

[Alteryx Designer](#)

[DataGrip](#)

[DBeaver](#)

[DbVisualizer](#)

[Domo](#)

[IBM Cognos Analytics](#)

[Looker](#)

[Microsoft Excel](#)

[Microsoft Excel PowerPivot](#)

[Microsoft Power BI](#)

[Microstrategy Workstation](#)

[Preset](#)

[SAP Business Objects](#)

[Superset](#)

[Tableau](#)

Drivers

[Arrow Flight SQL JDBC](#)

[Dremio JDBC \(Legacy\)](#)

[Arrow Flight SQL ODBC](#)

Developing Custom Applications

To create a connection to Dremio and run queries, you can use [Arrow Flight SQL JDBC](#). You can also use [PyArrow](#).

Client Encryption

Transport Layer Security (TLS) communication is supported for encrypting communication between JDBC client applications and Dremio servers. See the configuration of client TLS for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information.

Transport Layer Security (TLS) communication is enabled by default for Arrow Flight client applications. See the configuration of Arrow Flight encryption for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information. If you want to connect via unencrypted connections, you must explicitly disable `useEncryption` by setting it to `false` in the [connection parameters](#) for the Arrow Flight SQL ODBC driver.

Was this page helpful?

[Previous

Wikis and Tags](/current/data-products/govern/wikis-tags)[Next

Alteryx Designer](/current/client-applications/alteryx-designer)

Drivers

Developing Custom Applications

Client Encryption

Source:
<https://docs.dremio.com/current/client-applications/alteryx-designer>

Version: current [26.x]

On this page

Alteryx Designer

You can use Alteryx Designer to quickly prepare, blend, conform, and analyze data from datasets in Dremio.

Supported Versions

Alteryx Designer 10.6+

Prerequisites

Ensure that your operating system is 64-bit Windows 10 or later.

Download, install, and configure the [Arrow Flight SQL ODBC driver](#).

If you want to authenticate to Dremio by using a personal access token (PAT), rather than by using a password, generate a PAT. See [Personal Access Tokens](#) for the steps.

Selecting Dremio as a Data Source

In Alteryx Designer, select **File > New Workflow**.

Drag the **Input Data** tool from the tool palette on to the workflow canvas.

In the configuration properties for Input Data, click the arrow on the right side of the **Connect a File or Database** field.

In the Data connections dialog, follow these steps:

- a. Select **Recent** and click **Clear List** in the top-right corner if there are any entries on the page.
- b. Select **Data Sources**.

c. Scroll down to the option **Generic connection**.

d. Click either **ODBC** or **OleDB**.

If you clicked **ODBC**, follow these steps in the ODBC Connection dialog:

a. In the **Data Source Name** field, select the data source name for the Arrow Flight SQL ODBC driver.

b. Specify the username to use for the connection to Dremio.

c. Specify either a password or a personal access token to use with the username.

d. Click **OK**.

If you clicked **OleDB**, follow these steps in the Data Link Properties dialog:

a. On the **Provider** tab, select **Microsoft OLE DB Provider for ODBC Drivers**.

b. Click **Next>>**.

c. For step 1 on the **Connection** tab, select **Use data source name**, and then select the data source name for the Arrow Flight SQL ODBC driver.

d. For step 2 on the **Connection** tab, specify the username to use for connections to Dremio, then specify either a password or a personal access token to use with the username.

e. (Optional) Click **Test Connection** to find out whether the info you specified on this tab is correct.

f. Click **OK**.

You can now browse and query datasets that are in Dremio.

warning

If you are using an Arrow Flight SQL ODBC driver, it only supports a single connection, and to load multiple tables (or datasets), you should do it sequentially. Otherwise, if you try to do it in parallel, the driver raises an error.

Was this page helpful?

[Previous

Connect Client Applications](/current/client-applications/)[Next

Apache Superset](/current/client-applications/superset)

Supported Versions

Prerequisites

Selecting Dremio as a Data Source

Source:
<https://docs.dremio.com/current/client-applications/superset>

Version: current [26.x]

On this page

Apache Superset

You can use [Superset](#) to query and visualize data.

Supported Versions

Superset 1.5.3 and later

Dremio SQLAlchemy connector 3.0.2 and later

Supported Authentication Methods

Use the username and password of an account in your Dremio cluster.

Use the username of an account in your Dremio cluster and a personal access token (PAT) created in Dremio. To create a PAT, follow the steps in [Creating a PAT](#). After you obtain a PAT, it is recommended that you URL-encode it. To encode it locally on your system, you can follow these steps:

In a browser window, right-click an empty area of the page and select **Inspect** or **Inspect Element**, depending on your browser.

In the top bar of the inspection pane, click **Console**.

Type ``encodeURIComponent("<PAT>")``, where ``<PAT>`` is the personal access token. The URL-encoded PAT appears in red on the next line. You can highlight it and copy it to your clipboard.

Prerequisites

If you installed Superset according to [the instructions for installing from scratch](#), install the Dremio SQLAlchemy Connector on the system or in the VM where Apache Superset is running. Instructions are in the [sqlalchemy_dremio repository](#) in GitHub.

Creating a Connection

If you are using a version of Superset earlier than 2.1.0, follow these steps:

Select **Data > Databases** in the menu bar at the top of the screen.

Click the **Database** button in the top-right corner of the screen.

If you are using version 2.1.0 or later of Superset, follow these steps:

Click **Datasets** in the menu bar at the top of the screen.

Click the plus (+) icon in the top-right corner.

-

Select **Data > Connect database**.

In the **Connect a Database** dialog, follow these steps:

Select **Other** from the **Supported Databases** field.

In the **Display Name** field, name the new connection.

If you want to authenticate by using a username and password, specify in the **SQLAlchemy URI** field a URI that is in this format:

Format of URIs with username and password authentication

```
dremio+flight://<username>:<password>@<host>:<port>/<schema>[?option1=value[&,...]]
```

`<username>`: The username of the Dremio account to use.

`<password>`: The password of the Dremio account to use.

`<host>`: The hostname or IP address of the coordinator node of the Dremio cluster.

`<port>`: The port to connect to on the coordinator node. Unless explicitly changed on the node, the port is 32010.

`<schema>`: The name of the database schema to use by default when a schema is not given in a query. Providing a schema is optional. Specifying a schema does not prevent queries from being issued for other schemas; such queries must explicitly include the schema.

`[?option1=value[&,...]]`: One or more optional properties, separated by ampersands (`&`). See [SSL Connection Properties](#) and [Advanced Properties](#). Example URI with username and password authentication

```
dremio+flight://myUserID:myPassword@myHost:32010/Samples?UseEncryption=false
```

If you want to authenticate by using a personal access token, specify in the **SQLAlchemy URI** field a URI that is in this format:

Format of URIs with PAT authentication

```
dremio+flight://<username>:<PAT>@<host>:<port>/<schema>[?option1=value[&,...]]
```

`<username>`: The username of the Dremio account to use.

`<PAT>`: The URL-encoded personal access token that you obtained from Dremio Cloud. See [Supported Authentication Methods](#).

`<host>`: The hostname or IP address of the coordinator node of the Dremio cluster.

`<port>`: The port to connect to on the coordinator node. Unless explicitly changed on the node, the port is 32010.

`<schema>`: The name of the database schema to use by default when a schema is not given in a query. Providing a schema is optional. Specifying a schema does not prevent queries from being issued for other schemas; such queries must explicitly include the schema.

`^[/option=value[;...]]` : One or more optional properties, separated by semicolons. See SSL Connection Properties and Advanced Properties. Example URI with PAT authentication`

```
dremio+flight://myUserID:myPAT@myHost:32010/Samples?UseEncryption=false
```

Test the connection. If the test fails, check the syntax and values in the connection URI.

Click **Connect**.

SSL Connection Parameters

Use the following parameters to configure SSL encryption and verification methods:

Name	Type	Description	Default Value
UseEncryption	integer	Forces the client to use an SSL-encrypted connection to communicate with Dremio. Accepted values: `true`, the client communicates with Dremio by using SSL encryption; `false`, the client does not communicate with Dremio by using SSL encryption.	true
disableCertificateVerification	integer	Specifies whether to verify the host certificate against the trust store. Accepted values: `false`, verifies the certificate against the trust store; `true`, does not verify the certificate against the trust store.	false
trustedCerts	string	The full path of the .pem file containing certificates trusted by a CA, for the purpose of verifying the server. If this option is not set, defaults to using the trusted CA certificates .pem file. The TLS connection fails if you do not specify a value when UseEncryption is true and disableCertificateVerification is false.	N/A

Advanced Parameters

Name	Type	Description	Default Value
routing_queue	string	Specifies the queue to route queries to during a session. Direct Routing is used to specify the exact queue and execution cluster to run queries on for a given ODBC session. With Direct Routing, workload-management (WLM) rules are not considered; instead, queries are routed directly to the specified queue. For more information, see Workload Management .	N/A
routing_tag	string	When this parameter is set, the specified tag is associated with all queries executed within a session. Rules can check for the presence of a tag with the function "tag()". For more information, see Workload Management .	N/A

Was this page helpful?

[\[Previous](#)

[Alteryx Designer\]\(/current/client-applications/alteryx-designer\)\[Next](#)

[DataGrip\]\(/current/client-applications/datagrip\)](#)

[Supported Versions](#)

[Supported Authentication Methods](#)

Prerequisites

Creating a Connection

SSL Connection Parameters

Advanced Parameters

Source:
<https://docs.dremio.com/current/client-applications/datagrip>

Version: current [26.x]

On this page

DataGrip

You can run SQL from [DataGrip](#) to explore your data through Dremio. DataGrip supports connections to Dremio through the Arrow Flight SQL JDBC driver.

Supported Versions

Dremio connectivity is supported from DataGrip running on Windows, macOS, or Linux. It is recommended that you use the latest available version of DataGrip.

Supported Authentication Methods

You can authenticate your connection to Dremio using your Dremio username and password.

Prerequisites

Download the [Arrow Flight SQL JDBC driver](#).

Connecting to Dremio

Follow the steps below to connect to Dremio:

Create a project in DataGrip (see [Quick start with DataGrip](#) for more information).

Open the Database Explorer, click the **+** icon, then click **Driver and Data Source**.

[!Add Driver and Data Source](#)

Select the **Drivers** tab, then click **+** to add a new driver.

Fill in the following details for the new driver:

Name: Provide a name to identify the driver in DataGrip (e.g., Arrow Flight SQL 10).

Driver Files: Click **+**, click **Custom JARs...**, then select the Arrow Flight SQL driver (*flight-sql-jdbc-driver-10.0.0.jar*) from the location where you downloaded it.

Class: org.apache.arrow.driver.jdbc.ArrowFlightJdbcDriver

[!Driver Details](#)

At the bottom of the Data Sources and Drivers panel, click **Create Data Source**.

Ensure that the driver you just created is selected under **Project Data Sources**.

For Authentication, select **User & Password**, and provide the Dremio username and password to send for authentication.

For URL, follow the guidance under [Connecting to Databases](#).

The following is an example URL for a local Dremio installation that does not use an encrypted flight port:

Example Flight SQL URL

```
jdbc:arrow-flight-sql://localhost:32010?useEncryption=false
```

Click **Test Connection** to confirm a valid connection to Dremio.

[!Test Connection](#)

Click **OK** to save driver and data source.

Run a simple query to see how results are displayed in DataGrip.

[!Run a Query](#)

note

When querying tables and views in Dremio, ensure you are using the fully qualified path. For example, ``SELECT * FROM Samples."samples.dremio.com"."NYC-taxi-trips"``.

Was this page helpful?

[\[Previous](#)

[Apache Superset\]\(/current/client-applications/superset\)\[Next](#)

[DBEaver\]\(/current/client-applications/dbeaver\)](#)

[Supported Versions](#)

[Supported Authentication Methods](#)

[Prerequisites](#)

[Connecting to Dremio](#)

Source:
<https://docs.dremio.com/current/client-applications/dbeaver>

Version: current [26.x]

On this page

DBeaver

You can run SQL from [DBeaver](#) to explore your data in your data lakes and relational databases through Dremio and the Arrow Flight SQL JDBC driver.

note

If you want to use DBeaver with the legacy JDBC driver, see the instructions [here](#).

Supported Authentication Methods

You can use your Dremio username and password, or you can use a personal access token (PAT) that you obtained from Dremio.

Prerequisites

Download the [Arrow Flight SQL JDBC driver](#).

If you want to authenticate your connection to Dremio by using a personal access token, see [Creating a PAT](#) for the steps to obtain one.

Connecting to Dremio

In DBeaver, add the JDBC driver as a new driver. You need to do this only once, and can skip this step if DBeaver already lists this driver in its Driver Manager dialog:

- a. In the menubar, select **Database > Driver Manager**.
- b. In the Driver Manager dialog, click **New**.
- c. In the Settings section, follow these steps:

In the **Name** field, specify a name for the driver, such as "Arrow Flight SQL JDBC".

In the **Driver Type** field, ensure that **Generic** is the selected driver type.

In the **Class Name** field, specify `org.apache.arrow.driver.jdbc.ArrowFlightJdbcDriver`.

In the **URL Template** field, specify `jdbc:arrow-flight-sql://{host}:{port}`.

In the **Default Port** field, specify `32010`.

d. In the Libraries section, click **Add File** and select the `.jar` file for the Arrow Flight SQL JDBC.

e. Click **OK**.

Create a connection to Dremio that uses the driver:

- a. Select **Database > New Connection from JDBC URL**.

b. In the Create New Connection from JDBC URL dialog, type ``jdbc:arrow-flight-sql://<hostname>:32010``, where ``<hostname>`` is the hostname of your coordinator node. DBeaver lists the driver in the **Drivers** field.

c. Select the driver and click **Next**.

d. In the Connect to a Database dialog, provide your authentication credentials by following either of these sets of steps:

To use a personal access token that you obtained from Dremio:

In the **Username** field, specify the username for which the PAT was generated.

In the **Password** field, paste your personal access token.

To use your Dremio username and password:

In the **Username** field, specify your username.

In the **Password** field, your password.

e. If connections to Dremio Software will not be encrypted, add the ``useEncryption`` property as a driver property, and set the value to ``false``. The default for this property is ``true``.

f. (Optional) Click **Test Connection**. If the connection works, the **Connection Test** dialog opens and indicates that DBeaver is able to connect to Dremio. The connection is not held open. Click **OK**.

g. Click **Finish**.

Was this page helpful?

[Previous

DataGrip](/current/client-applications/datagrip)[Next

DbVisualizer](/current/client-applications/dbvisualizer)

Supported Authentication Methods

Prerequisites

Connecting to Dremio

Source:
<https://docs.dremio.com/current/client-applications/dbvisualizer>

Version: current [26.x]

On this page

DbVisualizer

DbVisualizer is a SQL runner that works with any JDBC-compliant data source. You can

run SQL from it to explore your data in your data lakes and relational databases through Dremio and the Arrow Flight SQL JDBC driver.

Supported Versions

You can use any version of DbVisualizer, as long as you use Dremio 21 or later.

Supported Authentication Methods

You can use your Dremio username and password, or you can use a personal access token (PAT) that you obtained from Dremio.

Prerequisites

Download the [Arrow Flight SQL JDBC driver](#).

If you want to authenticate your connection to Dremio by using a personal access token, see [Creating a PAT](#) for the steps to obtain one.

Connecting to Dremio

Add the Arrow Flight SQL JDBC driver to DbVisualizer's Driver Manager. You need to do this only once, and can skip this step if DbVisualizer already lists this driver in its Driver Manager dialog:

- a. Select **Tools > Driver Manager**.
- b. Above the **Driver Name** list of the **Driver Manager** dialog, click the plus (+) symbol.
- c. In the **Name** field, name the driver.
- d. Under **Driver artifacts and jar files**, click the plus icon, browse to the `.jar` file that you downloaded, select it, and click **Open**. DbVisualizer loads the `.jar` file.
- e. If you are not using TLS encryption for connections to Dremio, turn off encryption:

Click **Properties** next to **Driver Settings**.

Click the plus icon to add a new property.

Name the parameter `useEncryption` and set the value to `false`.

Click **Apply**.

- f. Close the **Driver Manager** dialog.

Create a connection to Dremio:

- a. In the menubar, select **Database > Create Database Connection**.
- b. Double-click **Custom** at the bottom of the **Driver Name** list.
- c. Name the connection.

d. In the **Settings Format** field, select **Database URL**.

e. Click in the **Driver Type** field and then double-click the name that you gave to the Arrow Flight SQL JDBC driver.

f. In the **Database URL** field, specify a URL in this format, where `host` is the hostname of your coordinator node: `jdbc:arrow-flight-sql://{host}:32010`

g. In the **Database Userid** and **Database Password** fields, specify your authentication credentials:

To use a personal access token that you obtained from Dremio:

In the **Database Userid** field, specify the username for which the PAT was generated.

In the **Database Password** field, paste your personal access token.

To use your Dremio username and password:

In the **Database Userid** field, specify your username.

In the **Database Password** field, your password.

h. If you are not using TLS encryptions for connections to Dremio, click the **Properties** tab and ensure that the property `useEncryption` is listed and that the value is `false`. Then, click the **Connection** tab.

f. Click **Finish**.

DbVisualizer creates the connection and opens it.

note

If you want to use DbVisualizer with the legacy JDBC driver, see the instructions [here](#).

Was this page helpful?

[Previous

DBeaver](/current/client-applications/dbeaver)[Next

Domo](/current/client-applications/domo)

Supported Versions

Supported Authentication Methods

Prerequisites

Connecting to Dremio

Source:
<https://docs.dremio.com/current/client-applications/domo>

Version: current [26.x]

On this page

Domo

Domo is a cloud-based platform designed to provide direct, simplified, real-time access to business data for decision makers across the company with minimal IT involvement.

Supported Authentication Methods

Use the username and password of an account in your Dremio cluster.

Use a username and a personal access token (PAT). To create one, see [Creating a PAT](#).

Creating a Cloud Integration with Dremio Software

Click the **Data** tab to open the Datasets page.

Click the **Federated** tab to open the **Amplify existing cloud warehouses** dialog.

Next to **Native integration**, click **Dremio**.

In the **Cloud integrations** dialog, click **Add new integration**.

In step 1 of the **Connect a Dremio cloud integration** wizard, follow these sub-steps:

In the **Integration name** field, specify a unique name for the integration.

(Optional) In the **Integration description** field, briefly describe the integration.

Select **Dremio Software** as the connection type.

Click **Next**.

In step 2 of the wizard, follow these sub-steps:

In the **Dremio connection URL** field, specify the following connection URL:

Connection URL

```
jdbc:dremio:direct=<hostname>:<port>;ssl=<true-or-false>
```

``ssl``: Specifies whether to encrypt communication with the Dremio cluster. Set to ``true`` only if encryption for communication with JDBC clients is configured in the cluster. See the configuration of client TLS for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information.

In the **Username** field, specify the username of the Dremio account that you want to use for authenticating to Dremio.

In the **Password** field, specify either the password for the Dremio account or a PAT.

Click **Next**.

Select the tables that you want to use with Domo through this integration.

Click **Create Datasets**.

Datasets are created from the tables, though no data is moved or copied. Datasets in Domo are connections to data in data sources.

Was this page helpful?

[Previous

DbVisualizer][current/client-applications/dbvisualizer][Next

IBM Cognos Analytics][current/client-applications/cognos)

Supported Authentication Methods

Creating a Cloud Integration with Dremio Software

Source:
<https://docs.dremio.com/current/client-applications/cognos>

Version: current [26.x]

On this page

IBM Cognos Analytics

You can run SQL from [Cognos Analytics](#) to explore your data through Dremio. Cognos Analytics Dynamic Query supports connections to Dremio through the Dremio JDBC driver.

Supported Versions

To find out which versions of Dremio are supported with IBM Cognos 11.2.x, see [DQM testing of vendor-supported client driver versions for each Cognos Analytics 11.2.x release](#).

To find out which versions of Dremio are supported with IBM Cognos 12.0.x, see [DQM testing of vendor-supported client driver versions for each Cognos Analytics 12.0.x release](#).

Supported Authentication Methods

You can use your Dremio username and password, a personal access token (PAT), or an access token from an identity provider that supports OpenID.

If you want to use a PAT, follow these steps before creating a connection to your Dremio cluster from Cognos:

Ensure that your Dremio administrator has followed the steps in [Enabling the Use of PATs](#).

[Create a PAT](#).

If you want to use an access token from an identity provider that supports OpenID,

ensure that your Dremio administrator has followed the steps in [OpenID Authentication](#).

Creating a Connection

Launch Cognos Analytics.

Navigate to **Manage > Data Server Connections**.

Click **Add Data Server** and select **Dremio** as the type of connection.

In the **JDBC URL** field, specify the URL for the Dremio coordinator by using this template:

JDBC URL template

```
jdbc:dremio:direct=<DREMIO_COORDINATOR>:31010[;schema=<OPTIONAL_SCHEMA>]
```

Follow one of these steps to configure a method for authenticating to Dremio:

If you want to connect to Dremio by using a username and a password, specify the username and password.

If you want to connect to Dremio by using a personal access token (PAT), specify ` \$token ` as the username and paste the PAT into the **Password** field.

If you want to connect to Dremio by using access tokens, select a Cognos namespace which has been configured to use OpenID Connect.

Save the connection definition.

Click **Test** to confirm that the connection succeeds.

Was this page helpful?

[Previous

Demo](/current/client-applications/domo)[Next

Looker](/current/client-applications/looker)

Supported Versions

Supported Authentication Methods

Creating a Connection

Source:
<https://docs.dremio.com/current/client-applications/looker>

Version: current [26.x]

On this page

Looker

You can use [Looker](#) to query and visualize data by means of Dremio.

Supported Authentication Methods

There are two methods of authenticating that you can choose from when you connect from Looker to Dremio:

Use a username and password for your Dremio cluster.

Use a personal access token (PAT) obtained from Dremio. To create a PAT, follow the steps in [Creating a PAT](#).

Creating a Connection

Log into Looker.

In the menu bar at the top of the page, select **Admin**, and then select **Connections** under **Database**.

Click the **Add Connection** button in the top-right corner of the page to open the Connection Settings page for creating a connection.

Specify a name for the connection.

In the **Dialect** field, select **Dremio 11+**.

In the **Remote Host:Port** fields, specify the hostname or IP address of your Dremio cluster, as well as the port to connect to. By default, the port number is 31010.

In the **Database** field, specify any value. Though Looker requires a value in this field, Dremio does not use the value.

In the **Username** and **Password** fields, specify your authentication credentials:

If you want to authenticate by using a Dremio username and a password, specify them in the **Username** and **Password** fields.

If you want to authenticate by using a personal access token, specify these values:

In the **Username** field, specify your Dremio username.

In the **Password** field, paste the personal access token.

If your Dremio cluster is configured to use TLS, ensure that the **SSL** check box is selected.

Click **Test These Settings** at the bottom of the page to check that the information that you specified is all valid.

Click **Add Connection** if the test of the connection is successful.

The new connection is listed on the Connections page.

Was this page helpful?

[\[Previous](#)

IBM Cognos Analytics](/current/client-applications/cognos)[Next

Microsoft Excel](/current/client-applications/microsoft-excel/)

Supported Authentication Methods

Creating a Connection

Source:
<https://docs.dremio.com/current/client-applications/microsoft-excel/>

Version: current [26.x]

On this page

Microsoft Excel

Supported Versions

warning

Microsoft Excel version 16.95 (25030928) for Mac introduces a change that causes Excel to crash when using the Arrow Flight SQL ODBC driver. The crash occurs immediately after authentication, before the query/source selection dialog appears. This issue only affects Excel 16.95 on macOS. Excel on Windows is not impacted. See [Hiding the SQL Tables Listing](#) for a fix.

Microsoft Excel in Microsoft 365

Prerequisites

Ensure that you are using Dremio v22.0 or later.

Ensure that your operating system is 64-bit Windows 10 or later.

Download, install, and configure the [Arrow Flight SQL ODBC driver](#).

If you want to authenticate to Dremio by using a personal access token (PAT), rather than by using a password, generate a PAT. See [Personal Access Tokens](#) for the steps.

Connecting to Dremio

In Excel, select **Data** from the menu bar.

Click **Get Data**.

Select **From Other Sources > From ODBC**.

In the From ODBC dialog, select the data source name that you specified when you configured the Arrow Flight SQL ODBC driver.

-

Specify the username to use for the connection to Dremio.

Specify either a password or a personal access token to use with the username

In the Navigator dialog, select a dataset.

Click **Load**.

Hiding the SQL Tables Listing

You can prevent Excel from crashing by enabling the ``hideSQLTablesListing`` flag to hide the list of available sources in the query/source selection dialog. This flag can be used for Mac computers with an Apple silicon or an Intel processor.

To set the configuration:

Go to the **System DSN** tab of the ODBC Manager.

Click **Configure**.

Change the value of the ``hideSQLTablesListing`` keyword to ``true``.

Click **OK**.

Using the Extended Flight SQL Buffer

Apple Silicon Not Supported

The Extended Flight SQL Buffer feature was designed for Apple Silicon processors, but the Arrow Flight SQL ODBC driver is not supported on Apple Silicon M1, M2, and M3 processors. This feature is no longer available.

Was this page helpful?

[Previous

Looker](/current/client-applications/looker)[Next

Microsoft Excel
PowerPivot](/current/client-applications/microsoft-excel/microsoft-excel-powerpivot)

Supported Versions

Prerequisites

Connecting to Dremio

Hiding the SQL Tables Listing

Using the Extended Flight SQL Buffer

Source:
<https://docs.dremio.com/current/client-applications/microsoft-power-bi/>

Version: current [26.x]

On this page

Microsoft Power BI

Connect [Microsoft Power BI](#) to visualize your data and create reports.

You can connect Power BI to Dremio in one of the following ways:

Configure a reusable connection to use in Power BI Desktop, Power BI Gateway, or Power BI Service. Power BI Service can connect to Dremio through DirectQuery or through Power BI Gateway.

Connect to a specific dataset by downloading the `.pbids` file from Dremio and opening it in Power BI Desktop.

Supported Authentication Methods

From Power BI, you can authenticate to Dremio with one of the following methods:

Username and password: Use your Dremio credentials.

Personal access token (PAT): For details, see [Personal Access Tokens](#).

Single sign-on (SSO) through OAuth 2.0: For steps on how to configure SSO, see [Enable SSO to Dremio from Power BI](#).

Connect to Dremio from Power BI

The Power BI connector for Dremio now supports connectivity through the open-source [Arrow Database Connectivity \(ADBC\) driver](#), which Dremio highly recommends using to connect to Dremio. To enable reports to use ADBC, see [Enable Connectivity with ADBC](#).

Existing connections will continue to work, but we recommend using the embedded ADBC driver for all new reports and migrating existing reports to ADBC to benefit from improved performance and supportability.

To connect to Dremio from Power BI Desktop:

Click **Get data**, search for `Dremio`, select **Dremio Software**, and click **Connect**.

In the Dremio Software dialog, follow these steps:

- Use the Flight SQL ADBC driver and in the **Server** field specify your Dremio hostname with the `adbc://` prefix. Example: `adbc://acme-company.dremio.com`
- (Optional) Complete the other fields in the dialog as you normally would.
- Under **Data Connectivity mode**, select either **Import** or **DirectQuery**. Click **OK**.
- For **Authentication Method**, select **Basic** or **Key**.

Basic: Enter your Dremio username and password.

Key: Paste in the personal access token you obtained from Dremio. For details, see [Personal Access Tokens](#).

Click **Connect**.

note

Creating Dataflows through Power BI Service is also supported. To create a dataflow, click **New > Dataflow**. For the data source connection, follow the steps above.

Create a Live Connection to a Dataset from Dremio

You can generate a Microsoft Power BI Data Source (`.pbids`) file that represents a live connection to a dataset that is in Dremio. No actual data is stored in this file, and you can think of it as a shortcut to a Power BI Desktop session with a preconfigured view of your data.

note

The `.pbids` file download option must be enabled for users to have access to this feature. To enable this feature, see [Enable the `.pbids` file download](#).

To create a live connection to a dataset:

In Dremio, navigate to the dataset.

Click the ellipsis (...) next to the dataset name.

Select **Download .pbids file**.

Open the downloaded file in Power BI Desktop.

Authenticate using your preferred method.

Power BI Gateway

To enable Power BI users to connect to Dremio via Power BI Gateway:

Install and configure [Power BI Gateway](#) on a machine that can connect to your Dremio cluster.

In the Power BI Gateway configuration, add Dremio as a data source using the same connection details as above.

Advanced Configuration

Enable Connectivity with ADBC

Dremio supports connectivity through Arrow Database Connectivity (ADBC). To enable this for Power BI Service, see the following options.

Enable the ADBC Option for a New Connection

In Power BI Desktop, click **Get data**.

In the Get Data dialog, locate and select **Dremio Software**, and click **Connect**.

In the Dremio Software dialog, in the **Server** field, specify your hostname with the `adbc://` prefix. Example: `adbc://acme-company.dremio.com`

(Optional) Complete the other fields in the dialog as you normally would.

Click **OK**.

Authenticate using your preferred method, and click **Connect**.

Enable the ADBC Option for an Existing Connection

In Power BI Desktop, go to **Data source settings**, select your source, and click **Change source**.

In the Dremio Software dialog, update the **Server** field by adding the `adbc://` prefix before the hostname. Example: `adbc://acme-company.dremio.com`. If you're unable to edit the source this way, click **Transform data**, then click **Advanced Editor** in the **Home** tab. In the dialog that appears, update the hostname/server with the `adbc://` prefix, and click **Done**.

Click **OK**.

Reauthenticate using your preferred method, and click **Connect**.

Enable the `.pbids` File Download in the Dremio Console

To enable the `.pbids` file download feature:

In Dremio, go to **Admin > Settings**.

In the **Support** section, enable **Allow downloading of .pbids files**.

Click **Save**.

Enable SSO to Dremio from Power BI

SSO is supported only for datasets that use DirectQuery.

note

SSO only works for reports created using the Dremio Cloud connector in Power BI Desktop. Reports created with the Dremio Software connector cannot use SSO by simply changing credentials, they must be converted first.

To convert existing reports from the Dremio Software connector to the Dremio Cloud connector, you'll need to modify the connection in Power BI's Advanced Editor to change the function from `Dremio.Databases` to `DremioCloud.DatabasesByServerV370`.

Prerequisites: Configure Dremio for Microsoft Entra ID

Before enabling SSO for Power BI reports, Dremio must be configured to use Microsoft Entra ID (Azure AD) as an identity provider.

Required configuration:

Set ``services.coordinator.web.auth.type`` to the Microsoft Entra ID / OIDC provider type configured for your deployment (for example, ``azuread``)

Provide the required OIDC or Azure AD configuration file (``azuread.json`` or equivalent)

Deploy configuration files to all coordinator nodes

Restart Dremio cluster after applying changes

For complete setup steps, see [Configure Microsoft Entra ID](#).

note

All authentication setup must be done through configuration files before enabling SSO in Power BI.

The following steps configure the Power BI side of SSO. These steps assume your Dremio deployment is already configured to authenticate users via Microsoft Entra ID.

To enable SSO for Power BI reports:

Enable SSO for a DirectQuery Report

To enable SSO for a report that uses DirectQuery:

In Power BI Service, open the workspace to which you published the report.

Find the dataset that is associated with the report, click the three dots next to its name, and select **Settings**.

In the settings for the dataset, expand **Data source credentials**.

Click **Edit credentials**.

For **Authentication method**, select **OAuth2**.

In the **Privacy level setting for this data source** field, ensure that **Private** is selected.

Select the check box **Report viewers can only access this data source with their own Power BI identities using DirectQuery**.

Click **Sign in**.

Enable SSO for Reports with Power BI Gateway

To enable SSO when you are using Power BI Gateway:

In Power BI Service, open the workspace to which you published the report.

Find the dataset that is associated with the report, click the three dots next to its name,

and select **Settings**.

In the settings for the dataset, expand **Gateway connection**.

Recreate your data source by following these steps:

Select the **Maps to** field.

Select **Manually add to gateway**.

For **Data Source Name**, enter a name for the data source.

For **Data Source Type**, select **Dremio Software**.

For **Server**, enter your Dremio hostname with the ``adbc://`` prefix. Example: ``adbc://acme-company.dremio.com``

For **Authentication Method**, select **OAuth2**.

Click **Add**.

In the **Data source credentials** section, click **Edit credentials**.

For **Authentication method**, select **OAuth2**.

In the **Privacy level setting for this data source** field, ensure that **Private** is selected.

Select the check box **Report viewers can only access this data source with their own Power BI identities using DirectQuery**.

Click **Sign in**.

note

SSO requires the OAuth2 authentication method. Basic authentication and personal access tokens do not support SSO when used through Power BI Gateway.

Arrow Database Connectivity (ADBC) Limitations

ADBC is not enabled by default. It must be enabled by the owner of the report.

NativeQuery is not supported.

Metadata calls are not cached.

Power BI Desktop occasionally caches errors that might affect future connection attempts until the cache is cleared.

Complex data types such as ``MAP`` and ``INTERVAL`` are not supported.

When using DirectQuery, chaining functions is supported, but some complex scenarios may not work as expected. Complex optional parameters for functions are not supported.

Troubleshoot Power BI

Cached Data Issues

If you have previously installed older versions of Power BI Desktop, cached data may interfere with the newer versions of the Flight SQL drivers resulting in connection errors.

Problem

For example, when using Flight SQL ADBC, cached connection data in Power BI could cause the following errors:

```
`ADBC: IOError [] [FlightSQL] [FlightSQL] unresolved address (Unavailable;
GetObjects(GetDBSchemas))`
```

```
`ADBC: IOError [] [FlightSQL] [FlightSQL] connection error: desc = "transport:
authentication handshake failed: credentials: cannot check peer: missing selected ALPN
property. If you upgraded from a grpc-go version earlier than 1.67, your TLS
connections may have stopped working due to ALPN enforcement. For more details,
see: https://github.com/grpc/grpc-go/issues/434" (Unavailable;
GetObjects(GetDBSchemas))`
```

Solution

Clear the Power BI Desktop cache and any cached data source permissions involving Dremio connections by following these steps:

Clear Power BI Desktop Caches.

In Power BI Desktop, go to **File > Options and Settings > Data Source Settings**.

Select **Global Permissions**.

Clear all cached connections by clicking **Clear All Permissions**, or select specific Dremio data sources and click **Clear Permissions**.

After completing these steps, try reconnecting to Dremio using the instructions above.

Large Result Sets

Problem

When fetching data from Dremio with ADBC you may see the following error:

```
`Unexpected error: [FlightSQL] grpc: received message larger than max (43747370 vs.
16777216) (ResourceExhausted; DoGet: endpoint 0: [])`
```

Solution

By default, the ADBC driver accepts only messages up to 16 MiB in size. This can be fixed by updating the Power BI M expression to customize the connection as follows:

```
let
```

```
Source = DremioCloud.DatabasesByServerV370("your-server-here", [  
    MaxMessageSize = 67108864 // 64 MiB  
])  
in  
Source
```

Replace `your-server-here` with your actual Dremio server address. The `MaxMessageSize` parameter sets the maximum message size in bytes (67108864 = 64 MiB).

Was this page helpful?

[Previous

Microsoft Excel
PowerPivot](/current/client-applications/microsoft-excel/microsoft-excel-powerpivot)[Ne
xt

Microstrategy Workstation](/current/client-applications/microstrategy)

Supported Authentication Methods

Connect to Dremio from Power BI

Create a Live Connection to a Dataset from Dremio

Power BI Gateway

Advanced Configuration

Enable Connectivity with ADBC

Enable the `.pbids` File Download in the Dremio Console

Enable SSO to Dremio from Power BI

Arrow Database Connectivity (ADBC) Limitations

Troubleshoot Power BI

Cached Data Issues

Large Result Sets

Source:
<https://docs.dremio.com/current/client-applications/microstrategy>

Version: current [26.x]

On this page

Microstrategy Workstation

MicroStrategy Workstation makes it easy to build compelling visualizations and interactive dossiers in a matter of minutes and then easily share those dossiers with others.

Supported Versions

Microstrategy Workstation 2021 Update 9 and later.

Supported Authentication Methods

Use a Dremio username and password.

Use a Dremio username and a personal access token (PAT) that you obtained from Dremio.

Prerequisites

Install the latest version of Dremio's legacy JDBC driver:

Download the driver.

Move the `.jar` file to one of these locations:

On Windows: `C:\Program Files (x86)\Common Files\MicroStrategy\JDBC\`

On Linux: `/opt/mstr/MicroStrategy/install/JDBC\`

If you want to authenticate your connection to Dremio by using a PAT, see Creating a PAT for the steps to obtain one.

In Microstrategy Workstation, select **Help > Enable New Data Import Experience**.

Creating an Environment-level Integration with Dremio

You can create an integration with Dremio that can be used in more than one dossier.

In Microstrategy Workstation, connect to the environment that you plan to use.

Under **Administration** on the left side of the screen, click the + sign to the right of **Data Sources**.

In the Data Source Types dialog, click **Dremio**.

In the Add Data Source - Dremio dialog, follow these steps:

In the **Name** field, specify a name for the integration.

(Optional) In the **Description** field, describe the integration.

In the **Database Version** field, ensure that **Dremio** is selected.

In the **Default Database Connection** field, either select an existing connection to a Dremio cluster, or select **Add New Database Connection**.

If you selected **Add New Database Connection**, follow these steps in the Create New Database Connection dialog:

In the **Name** field, specify a name for the connection.

In the **Basic** section, follow these steps:

In the **Host Name** field, specify the IP address or hostname of the coordinator node of the Dremio cluster that you want to use this connection with.

In the **Port Number** field, specify the port number to use. The default port number is 31010.

In the **Authentication Mode** field, ensure that **Standard** is selected.

In the **Default Database Login** field, either select an existing set of authentication credentials for the Dremio Cluster, or click **Add New Database Login** to add a new set.

If you clicked **Add New Database Login** in the previous step, specify the name to associate with the credentials, and then follow either of these steps:

Specify the username and password of the Dremio account to use.

Specify the username of the Dremio account to use and a PAT, which you can paste into the **Password** field.

(Optional) In the **Advanced** section, set additional properties for Microstrategy Workstation to use when connecting to Dremio. See [Optional Advanced JDBC Driver Properties](#) for a list of the properties that you can use.

(Optional) In the **Security** section, toggle on the **Use TLS Encryption** switch if the Dremio cluster is configured to encrypt communication between it and JDBC clients. For more information, see the configuration of client TLS for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#).

(Optional) In the **Properties** section, specify non-default values for various properties of the connection.

Click **Save**.

In the Add Data Source - Dremio dialog, follow these steps:

(Optional) Click **Test** to find out whether Microstrategy Workstation can connect to the Dremio cluster by using the database connection. If the test fails, ensure that the connection is configured correctly.

Click **Save**.

You can now select this database connection when you create dossiers.

Was this page helpful?

[\[Previous](#)

[Microsoft Power BI\]\(/current/client-applications/microsoft-power-bi/\)](#)[\[Next](#)

[Preset\]\(/current/client-applications/preset\)](#)

Supported Versions

Supported Authentication Methods

Prerequisites

Creating an Environment-level Integration with Dremio

Source:
<https://docs.dremio.com/current/client-applications/preset>

Version: current [26.x]

On this page

Preset

You can use [Preset](#), a cloud service for Superset, to query and visualize data.

Supported Authentication Methods

Use the username and password of an account in your Dremio cluster.

Use the username of an account in your Dremio cluster and a personal access token (PAT) created in Dremio. To create a PAT, follow the steps in [Creating a PAT](#). After you obtain a PAT, it is recommended that you URL-encode it. To encode it locally on your system, you can follow these steps:

In a browser window, right-click an empty area of the page and select **Inspect** or **Inspect Element**, depending on your browser.

In the top bar of the inspection pane, click **Console**.

Type ``encodeURIComponent("<PAT>")``, where ``<PAT>`` is the personal access token. The URL-encoded PAT appears in red on the next line. You can highlight it and copy it to your clipboard.

Creating a Connection

Click **Settings** in the top-right corner, and select **Database Connections** under **Data**.

Click the **+Database** button in the top-right corner.

Select **Other** from the **Supported Databases** field of the Connect a Database dialog.

In the **Display Name** field, specify any name you prefer.

In the **Connect a Database** dialog, follow these steps:

Select **Other** from the **Supported Databases** field.

In the **Display Name** field, name the new connection.

If you want to authenticate by using a username and password, specify in the **SQLAlchemy URI** field a URI that is in this format:

Format of URIs with username and password authentication

```
dremio://<username>:<password>@<host>:<port>[/option=value[;...]]
```

`<username>`: The username of the Dremio account to use.

`<password>`: The password of the Dremio account to use.

`<host>`: The hostname or IP address of the coordinator node of the Dremio cluster.

`<port>`: The port to connect to on the coordinator node. Unless explicitly changed on the node, the port is 31010.

`[/option=value[;...]]`: One or more optional properties, separated by semicolons. See [SSL Connection Properties](#) and [Advanced Properties](#). Example URI with username and password authentication

```
dremio://myUserID:myPassword@myHost:31010/ssl=true;schema=Samples;routing_tag=thisTag
```

If you want to authenticate by using a personal access token, specify in the **SQLAlchemy URI** field a URI that is in this format:

Format of URIs with PAT authentication

```
dremio://<username>:<PAT>@<host>:<port>[/option=value[;...]]
```

`<username>`: The username of the Dremio account to use.

`<PAT>`: The personal access token to use.

`<host>`: The hostname or IP address of the coordinator node of the Dremio cluster.

`<port>`: The port to connect to on the coordinator node. Unless explicitly changed on the node, the port is 31010.

`[/option=value[;...]]`: One or more optional properties, separated by semicolons. See [SSL Connection Properties](#) and [Advanced Properties](#). Example URI with PAT authentication

```
dremio://myUserID:myPAT@myHost:31010/ssl=true;schema=Samples;routing_tag=thisTag
```

Test the connection. If the test fails, check the syntax and values in the connection URI.

Click **Connect**.

SSL Connection Properties

Use the following properties to configure SSL encryption and verification methods:

Name	Type	Description	Default Value
UseEncryption	integer	Forces the client to use an SSL-encrypted connection to communicate with Dremio. Accepted values: `true`, the client communicates with Dremio by using SSL encryption; `false`, the client does not communicate with Dremio by using SSL encryption.	true
disableCertificateVerification	integer	Specifies whether to verify the host certificate against the trust store. Accepted values: `false`, verifies the certificate against the trust store; `true`, does not verify the certificate against the trust store.	false
trustedCerts	string	The full path of the .pem file containing certificates trusted by a CA, for the purpose of verifying the server. If this option is not set, defaults to using the trusted CA certificates .pem file. The TLS connection fails if you do not specify a value when UseEncryption is true and disableCertificateVerification is false.	N/A

Advanced Properties

Name	Type	Description	Default Value
routing_queue	string	Specifies the queue to route queries to during a session. Direct Routing is used to specify the exact queue and execution cluster to run queries on for a given session. With Direct Routing, workload-management (WLM) rules are not considered; instead, queries are routed directly to the specified queue. For more information, see Workload Management .	N/A
routing_tag	string	When this property is set, the specified tag is associated with all queries executed within a session. Rules can check for the presence of a tag with the function "tag()". For more information, see Workload Management .	N/A

Was this page helpful?

[\[Previous](#)

[Microstrategy Workstation\]\(/current/client-applications/microstrategy\)\[Next](#)

[SAP Business Objects\]\(/current/client-applications/business-objects\)](#)

[Supported Authentication Methods](#)

[Creating a Connection](#)

[SSL Connection Properties](#)

[Advanced Properties](#)

Source:
<https://docs.dremio.com/current/client-applications/business-objects>

Version: current [26.x]

On this page

SAP Business Objects

Prerequisites

Dremio Business Objects integration requires:

SAP Business Objects 4.0+

Download, install, and configure the [Arrow Flight SQL ODBC driver](#).

Connecting to a Dremio cluster using Information Design Tool

Open Information Design Tool and a new project.

Create a new Relational Connection using the Generic ODBC3 datasource driver.

Select the Arrow Flight SQL ODBC DSN and test the connection.

Dremio schemas and tables are now available.

Using Dremio datasets in Web Intelligence Reports

In Information Design Tool, publish the Dremio connection to a repository.

Create a new Data Foundation.

Create a new Business Layer.

Publish the universe to a repository.

Open a web browser, go to Web Intelligence tool and select the published universe.

Configure the query.

Your Dremio dataset is ready to be used in Web Intelligence.

Was this page helpful?

[\[Previous](#)

[Preset\]\(/current/client-applications/preset\)\[Next](#)

[Tableau\]\(/current/client-applications/tableau\)](#)

Prerequisites

Connecting to a Dremio cluster using Information Design Tool

Using Dremio datasets in Web Intelligence Reports

Source:
<https://docs.dremio.com/current/client-applications/tableau>

Version: current [26.x]

On this page

Tableau

Connect [Tableau](#) to Dremio to derive powerful insights from your data and create real-time dashboards.

You can connect your Tableau application to Dremio in one of two ways:

Configure a reusable connection in Tableau Desktop, Tableau Server, or Tableau Cloud.

Connect to a specific dataset by downloading the `.tds`` file from Dremio and opening it in Tableau Desktop.

Supported Tableau Versions

Product	Supported Versions
Tableau Desktop	2022.1 and later
Tableau Server	2022.1 and later
Tableau Cloud	Latest version deployed by Tableau

Supported Authentication Methods

From Tableau, you can authenticate to Dremio with a username and password, or with a [personal access token \(PAT\)](#) that can be obtained from the Dremio console.

You can also configure single sign-on (SSO) through OAuth 2.0. For steps on how to configure SSO, see [Enabling SSO to Dremio from Tableau](#).

Tableau Desktop

Tableau Desktop includes a native connector that you can use to connect to Dremio.

Prerequisites for Using the Dremio JDBC Driver (Legacy)

To connect to Dremio, you'll also need to install the Dremio JDBC driver. Download the Dremio JDBC driver and copy it to Tableau Desktop's `Drivers`` folder.

Download driver for macOS by running this command in a Terminal window

```
curl -L https://download.dremio.com/jdbc-driver/dremio-jdbc-driver-LATEST.jar -o ~/Library/Tableau/Drivers/dremio-jdbc-driver-LATEST.jar
```

Download driver for Windows by running this command in a PowerShell window

```
Invoke-WebRequest "https://download.dremio.com/jdbc-driver/dremio-jdbc-driver-LATEST.jar" -Uri "C:\Program Files\Tableau\Drivers\dremio-jdbc-driver-LATEST.jar" -OutFile
```

Prerequisites for Using the Arrow Flight SQL JDBC Driver

The Tableau Desktop 2025.1+ connector for Dremio supports Arrow Flight SQL JDBC in place of the Dremio JDBC driver (Legacy). To change the driver, download the Arrow Flight SQL JDBC driver, copy it to Tableau Desktop's `Drivers` folder, and select the **Use Arrow Flight SQL Driver (preview)** option in the **Advanced** tab of the connection dialog.

Download driver for macOS by running this command in a Terminal window

```
curl -L https://repo1.maven.org/maven2/org/apache/arrow/flight-sql-jdbc-driver/18.3.0/flight-sql-jdbc-driver-18.3.0.jar -o ~/Library/Tableau/Drivers/flight-sql-jdbc-driver-18.3.0.jar
```

Download driver for Windows by running this command in a PowerShell window

```
Invoke-WebRequest -Uri "https://repo1.maven.org/maven2/org/apache/arrow/flight-sql-jdbc-driver/18.3.0/flight-sql-jdbc-driver-18.3.0.jar" -OutFile "C:\Program Files\Tableau\Drivers\flight-sql-jdbc-driver-18.3.0.jar"
```

Steps for Connecting

To create a Dremio source in Tableau Desktop:

Open Tableau Desktop. Under the **To a Server** section in the **Connect** panel, click **More**.

Select **Dremio**. The **Dremio** connection dialog opens.

In the connection dialog, for the **Product** field, select **Dremio Software**.

For the **Server** field, specify the hostname or IP address of your Dremio coordinator node.

In the **Port** field, specify the port, if it differs from the default port, which is `31010`.

In the **Authentication** field, select **Username and Password** or **OAuth 2.0**.

If you selected **Username and Password**, in the **Username** and **Password** fields, specify your Dremio credentials. If you have a personal access token, specify your username and then paste the token into the **Password** field.

If you selected **OAuth 2.0**, specify one of these URLs in the **Dremio Authorization Server** field, replacing `` with the hostname or IP address for your Dremio coordinator node:

If your Dremio cluster does not use SSL: `http://<dremio-host>:9047`

If your Dremio cluster does use SSL: `https://<dremio-host>:9047`

(Optional) If your Dremio cluster is configured for secure connections, select the **Require SSL** option.

(Optional for Tableau 2025.1+) If you are using the Arrow Flight SQL JDBC driver, in the **Advanced** tab, select the **Use Arrow Flight SQL Driver (preview)** option. Ensure that you have the Arrow Flight SQL JDBC driver downloaded.

(Optional) In the **Advanced** tab, specify the **Engine**, **Queue**, and **Tag**. For information about how these values are used, see [Workload Management](#).

Click **Sign In**.

Creating a Live Connection to a Dataset from Dremio

You can generate a Tableau Datasource (`.tds`) file that represents a live connection to a dataset that is in Dremio. No actual data is stored in this file, and you can think of it as a shortcut to a Tableau session with a preconfigured view of your data.

note

The `.tds` file download option must be enabled for users to have access to this feature. To enable this feature, see [Enabling the .tds file download](#).

To download a `.tds` file:

On the Datasets page in Dremio, find the dataset you want to work with and open the Details panel for the dataset.

Click the button that displays the Tableau logo. Dremio downloads a `.tds` file to your system.

Open the `.tds` file.

Authenticate using your username and password. To authenticate using SSO, follow these steps:

Sign into your identity provider. You are taken to the sign-in screen only the first time that you log into Dremio during a session in Tableau.

Click **Accept** in the Authorize App dialog. This dialog appears only the first time that you authenticate from Tableau through your identity provider.

Tableau Server

Tableau Server includes a native connector that you can use to connect to Dremio.

Prerequisites for Using the Dremio JDBC Driver (Legacy)

To connect to Dremio, you'll need to install the Dremio JDBC driver. Download the Dremio JDBC driver and copy it to the `Drivers` folder.

Download driver for Windows by running this command in a PowerShell window

```
Invoke-WebRequest -Uri  
"https://download.dremio.com/jdbc-driver/dremio-jdbc-driver-LATEST.jar" -OutFile  
"C:\Program Files\Tableau\Drivers\dremio-jdbc-driver-LATEST.jar"
```

Download driver for Linux by running this command in a command-line window

```
curl -L https://download.dremio.com/jdbc-driver/dremio-jdbc-driver-LATEST.jar -o /opt/tableau/tableau_driver/jdbc/dremio-jdbc-driver-LATEST.jar
```

Prerequisites for Using the Arrow Flight SQL JDBC Driver

The Tableau Server 2025.1+ connector for Dremio supports Arrow Flight SQL JDBC in place of the Dremio JDBC driver (Legacy). To change the driver, download the Arrow Flight SQL JDBC driver, copy it to the `Drivers` folder, and select the **Use Arrow Flight SQL Driver (preview)** option in the **Advanced** tab of the connection dialog.

Download driver for Windows by running this command in a PowerShell window

```
Invoke-WebRequest -Uri "https://repo1.maven.org/maven2/org/apache/arrow/flight-sql-jdbc-driver/16.1.0/flight-sql-jdbc-driver-16.1.0.jar" -OutFile "C:\Program Files\Tableau\Drivers\flight-sql-jdbc-driver-16.1.0.jar"
```

Download driver for Linux by running this command in a command-line window

```
curl -L https://repo1.maven.org/maven2/org/apache/arrow/flight-sql-jdbc-driver/16.1.0/flight-sql-jdbc-driver-16.1.0.jar -o /opt/tableau/tableau_driver/jdbc/flight-sql-jdbc-driver-16.1.0.jar
```

Steps for Connecting

To create a Dremio source in Tableau Server:

In a web browser, navigate to your Tableau Server site.

In your workbook, click **Add a Data Source**. Alternatively, you can [publish an existing data source](#) to Tableau Server.

In the **Connect to Data** dialog, select **Dremio** under the **Connectors** tab.

In the connection dialog, for the **Product** field, select **Dremio Software**.

For the **Server** field, specify the hostname or IP address of your Dremio coordinator node.

For **Port**, enter the port if it differs from the default `31010` port.

In the **Authentication** field, select **Username and Password** or **OAuth 2.0**.

If you selected **Username and Password**, in the **Username** and **Password** fields, specify your Dremio credentials. If you have a personal access token, specify your username and then paste the token into the **Password** field.

If you selected **OAuth 2.0**, specify one of these URLs in the **Dremio Authorization Server** field, replacing `` with the hostname or IP address for your

Dremio coordinator node:

If your Dremio cluster does not use SSL: `http://<dremio-host>:9047`

If your Dremio cluster does use SSL: `https://<dremio-host>:9047`

(Optional) If your Dremio cluster is configured for secure connections, select the **Require SSL** option.

(Optional for Tableau 2025.1+) If you are using the Arrow Flight SQL JDBC driver, in the **Advanced** tab, select the **Use Arrow Flight SQL Driver (preview)** option. Ensure that you have the Arrow Flight SQL JDBC driver downloaded.

(Optional) In the **Advanced** tab, you can specify the **Engine**, **Queue**, and **Tag**.

Click **Sign In**.

Tableau Cloud

Tableau Cloud includes a native connector that you can use to connect to Dremio.

note

The Tableau Cloud 2025.1 connector for Dremio has an option to use the [Arrow Flight SQL JDBC](#) driver in place of the Dremio JDBC driver to power the connection to Dremio. In the **Advanced** tab, select the **Use Arrow Flight SQL Driver (preview)** option.

Steps for Connecting

To create a Dremio source in Tableau Cloud:

In a web browser, navigate to your Tableau Cloud site.

Click **New > Published Data Source** to create a reusable data source or **Data > Add a Data Source** from within a workbook. Alternatively, you can [publish an existing data source](#) to Tableau Cloud.

In the **Connect to Data** dialog, select **Dremio** under the **Connectors** tab.

In the connection dialog, for the **Product** field, select **Dremio Software**.

For the **Server** field, specify the hostname or IP address of your Dremio coordinator node.

In the **Port** field, enter the port if it differs from the default `31010` port.

In the **Authentication** field, select **Username and Password** or **OAuth 2.0**.

If you selected **Username and Password**, in the **Username** and **Password** fields, specify your Dremio credentials. If you have a personal access token, specify your username and then paste the token into the **Password** field.

If you selected **OAuth 2.0**, specify one of these URLs in the **Dremio Authorization Server** field, replacing `<dremio-host>` with the hostname or IP address for your Dremio coordinator node:

If your Dremio cluster does not use SSL: `http://<dremio-host>:9047`

If your Dremio cluster does use SSL: `https://<dremio-host>:9047`

(Optional for Tableau 2025.1+) If you are using the Arrow Flight SQL JDBC driver, in the **Advanced** tab, select the **Use Arrow Flight SQL Driver (preview)** option.

(Optional) If your Dremio cluster is configured for secure connections, select the **Require SSL** option.

(Optional) In the **Advanced** tab, you can specify the **Engine**, **Queue**, and **Tag**.

Click **Sign In**.

If you're authenticating using SSO (OAuth 2.0), follow these steps:

Sign into your identity provider. You are taken to the sign-in screen only the first time that you log into Dremio during a session in Tableau Cloud.

Click **Accept** in the Authorize App dialog. This dialog appears only the first time that you authenticate from Tableau Cloud through your identity provider.

Advanced Configuration

Enabling the `.tds` File Download in the Dremio console

`ADMIN` privileges are required to make updates to this setting.

To enable users to download `.tds` files for datasets in Dremio, follow these steps:

Click the Settings icon in the left sidebar of a project.

Select **Project Settings**.

Select **BI Applications**.

Select the **Tableau** tab.

Toggle the **Enable Tableau Desktop** setting on.

After the organization administrator completes these steps, refresh your browser window.

Enabling SSO to Dremio from Tableau Enterprise

SSO using OAuth 2.0 is supported by Tableau Desktop 2022.3 or later, Tableau Server, and Tableau Cloud.

Users of Tableau Desktop will use SSO authentication whether connecting directly to Dremio or connecting through a `.tds` file downloaded from Dremio. If you want to use SSO to authenticate when connecting to Dremio through a `.tds` file, ensure that SSO is enabled and configured for your Dremio cluster before the file is downloaded.

To enable SSO to Dremio from Tableau, ensure that your Dremio cluster has SSO

configured with [Microsoft Entra ID](#) or an [OIDC identity provider](#) and follow these steps:

For Tableau Server only, follow the configuration steps.

Follow the steps to enable SSO to Dremio from Tableau.

Configuring SSO for Tableau Server

To configure SSO using [OAuth for Tableau Server](#), follow these steps:

Run the following command in the Tableau Services Manager (TSM) command line. Set a value for the `<tableau-server-domain-name-or-ip>` parameter, which is the domain name or IP of your Tableau Server deployment:

Configure OAuth for Tableau Server

```
tsm configuration set -k oauth.config.clients -v "[{\"oauth.config.id\":\"dremio\",
\"oauth.config.client_id\":\"https:\\/\\/connectors.dremio.app\\/tableau\",
\"oauth.config.client_secret\":\"test-client-secret\",
\"oauth.config.redirect_uri\":\"https://<tableau-server-domain-name-or-ip>/auth/add_oauth_token\"}]" --force-keys
```

To apply the changes to Tableau Server, run the command `tsm pending-changes apply`.

Configuring Dremio

To enable SSO authentication to Dremio from Tableau:

In the Dremio console, click the Settings icon and select the BI Applications page.

On the BI Applications page, click **Tableau**.

Ensure that **Enable single sign-on for Tableau** is toggled on.

For Tableau Server only: In the **Redirect URIs** field, paste in the redirect URI for your Tableau Server. If you have set up more than one Tableau Server, you can add multiple URIs, separating them with commas. Each URI uses this format, where `<tableau-server>` is the hostname or IP address of Tableau Server:

Redirect URI for Tableau Server

```
https://<tableau-server>/auth/add_oauth_token
```

Configuring an Identity Provider

Register an additional redirect URI: `https://<dremio-host>:xxxx/oauth/callback` or `http://<dremio-host>:xxxx/oauth/callback` in the SSO application configured in your identity provider. See the configuration instructions for [Microsoft Entra ID](#) or [OpenID Identity Providers](#) for additional information.

Customizing the Connection String

To add JDBC parameters to the JDBC URL that Tableau generates for connections to Dremio using the parameters from the connection dialog, see [Use a Properties file to customize a JDBC connection](#) in the Tableau documentation.

Manually Installing the Dremio Connector

If you are previewing a feature that hasn't been released or you have been provided a `.taco`` file with a fix that hasn't been released, you can manually install this version of the Dremio connector for temporary use.

To manually install the connector:

Download the `dremio.taco`` file.

Move the `dremio.taco`` file:

Copy dremio.taco file on macOS

```
cp <download-location>/dremio.taco ~/Library/Tableau/Connectors/
```

Copy dremio.taco file on Windows

```
copy C:\<download-location>\dremio.taco "C:\Program Files\Tableau\Connectors"
```

Move dremio.taco file for Linux (Tableau Server only)

```
mv <download-location>/dremio.taco /opt/tableau/connectors/dremio.taco
```

(Optional) If a new version of the Dremio JDBC driver is required, download it and copy it to Tableau Desktop's `Drivers`` folder by running the following command:

Download driver for macOS

```
curl https://download.dremio.com/jdbc-driver/dremio-jdbc-driver-LATEST.jar -o -l  
~/Library/Tableau/Drivers/dremio-jdbc-driver-LATEST.jar
```

Download driver for Windows

```
Invoke-WebRequest -Uri  
"https://download.dremio.com/jdbc-driver/dremio-jdbc-driver-LATEST.jar" -OutFile  
"C:\Program Files\Tableau\Drivers\dremio-jdbc-driver-LATEST.jar"
```

For Linux, download the driver from the [download site](#) and move it by using this command:

Download driver for Linux (Tableau Server only)

```
mv <download-location>/dremio-jdbc-driver-LATEST.jar  
/opt/tableau/tableau_driver/jdbc/dremio-jdbc-driver-LATEST.jar
```

Now you can connect to Dremio from Tableau Desktop or Tableau Server.

Exporting a Dremio Dataset with SSL Placeholder

If you have SSL enabled on Dremio, and you want to export a `.tds` file to use in a Tableau application for quickly connecting to a dataset, you can do so by adding the support key `export.tableau.extra-native-connection-properties` and set the value to `SSL=true`. This property allows you to set the JDBC connection string when exporting a `.tds` file. The default is an empty string. This is the only property currently supported for `export.tableau.extra-native-connection-properties`, and `true` is the only supported value.

To know how to set the `export.tableau.extra-native-connection-properties` support key, see [Support Keys](#).

Changing the Hostname

You can use the `export.bi.hostname` support key to change the default hostname of the SQL endpoint for generating TDS files.

To know how to set the `export.bi.hostname` support key, see [Support Keys](#).

Example: SSL setting

In the following example, SSL is enabled in the **dremio.conf** file. See [Using Wire Encryption](#) for more information.

Example SSL settings for generating a self-signed certificate with JDBC Dremio config

```
services.coordinator.client-endpoint.ssl.enabled: true
services.coordinator.client-endpoint.ssl.auto-certificate.enabled: true
```

Example: export.tableau.extra-native-connection-properties value

Example SSL property value

```
SSL=true
```

Limitations

When using Tableau with Dremio, avoid using periods in space or dataset names. Due to differences in hierarchy support, periods in paths are treated as separators, resulting in errors when navigating or selecting spaces or datasets with periods in their names.

Was this page helpful?

[\[Previous](#)

[SAP Business Objects](#)[(/current/client-applications/business-objects)][Next

[ThoughtSpot](#)[(/current/client-applications/thoughtspot)

Supported Tableau Versions

Supported Authentication Methods

Tableau Desktop

Prerequisites for Using the Dremio JDBC Driver (Legacy)

Prerequisites for Using the Arrow Flight SQL JDBC Driver

Steps for Connecting

Creating a Live Connection to a Dataset from Dremio

Tableau Server

Prerequisites for Using the Dremio JDBC Driver (Legacy)

Prerequisites for Using the Arrow Flight SQL JDBC Driver

Steps for Connecting

Tableau Cloud

Steps for Connecting

Advanced Configuration

Enabling the ``.tds`` File Download in the Dremio console

Enabling SSO to Dremio from Tableau Enterprise

Customizing the Connection String

Manually Installing the Dremio Connector

Exporting a Dremio Dataset with SSL Placeholder

Limitations

Source:
<https://docs.dremio.com/current/client-applications/thoughtspot>

Version: current [26.x]

On this page

ThoughtSpot

You can use [ThoughtSpot](#) to search directly against your data in Dremio for live analytics and actionable insights.

Supported Versions

Dremio supports ThoughtSpot cloud 8.3 and ThoughtSpot software 7.2.1.

Supported Authentication Methods

You can use your Dremio username and password.

Creating a Connection

note

While you're using the connection, the data fields that you create, modify, and delete in Dremio are reflected as table columns in ThoughtSpot. To account for new or outdated fields, you will need to go back into the data connection to check or uncheck the columns that you want added or removed on the Select Tables page.

Log into ThoughtSpot.

Go to **Data > Connections > Add Connection**.

On the Choose Your Data Warehouse page, specify your data connection details:

- a. In the **Name your connection** field, enter a name.
- b. (Optional) In the **Describe your connection** field, enter a brief description.
- c. For the **Choose your data warehouse** field, select **Dremio**.

Click **Continue**.

On the Dremio Connection Details page, specify your account credentials:

- a. To provide your Dremio username and password for authentication, select **Use Service Account**.
- b. In the **Host** field, enter the IP address for one of the coordinator nodes in your cluster.
- c. In the **Port** field, enter 31010.
- d. In the **User** field, enter your username.
- e. In the **Password** field, enter your password.

Click **Continue**.

On the Select Tables page, you can see all the data tables and views from Dremio. To select tables and columns from that list, select a table and check the boxes next to the columns for that table.

Click **Create Connection**.

In the **Create Connection** dialog, click **Confirm**.

Was this page helpful?

[\[Previous](#)

[Tableau\]\(/current/client-applications/tableau\)\[Next](#)

[Drivers\]\(/current/client-applications/drivers/\)](#)

[Supported Versions](#)

[Supported Authentication Methods](#)

[Creating a Connection](#)

Source:
<https://docs.dremio.com/current/client-applications/drivers/>

Version: current [26.x]

Drivers

Dremio provides Arrow Flight SQL ODBC and JDBC drivers:

[Arrow Flight SQL ODBC](#)

[Arrow Flight SQL JDBC](#)

Dremio also supports the [Dremio JDBC driver \(Legacy\)](#).

Was this page helpful?

[\[Previous](#)

[ThoughtSpot\]\(/current/client-applications/thoughtspot\)\[Next](#)

[Arrow Flight SQL JDBC\]\(/current/client-applications/drivers/arrow-flight-sql-jdbc-driver\)](#)

Source:
<https://docs.dremio.com/current/client-applications/dbvisualizer-legacy>

Version: current [26.x]

On this page

DbVisualizer

You can use the legacy Dremio JDBC driver to run [DbVisualizer](#), a SQL runner that works with any JDBC-compliant data source.

Supported Versions

You can use any version of DbVisualizer, if you use the legacy Dremio JDBC Driver 14.0.0 or later.

Supported Authentication Methods

You can use your Dremio username and password, or you can use a personal access token (PAT) that you obtained from Dremio.

Prerequisites

Download the [legacy Dremio JDBC driver](#).

If you want to authenticate your connection to Dremio by using a personal access token, see [Creating a PAT](#) for the steps to obtain one.

Connecting to Dremio

Add the legacy Dremio JDBC Driver to DbVisualizer's Driver Manager. You need to do this only once, and can skip this step if DbVisualizer already lists this driver in its Driver Manager dialog:

- a. In the menubar, select **Tools > Driver Manager**.
- b. In the **Driver Name** list of the **Driver Manager** dialog, select **Dremio**.
- c. Click the folder icon to find and select the downloaded Dremio JDBC driver.
- d. Close the **Driver Manager** dialog.

Create a connection to Dremio:

- a. In the menubar, select **Database > Create Database Connection**.
- b. Double-click **Dremio** in the **Driver Name** list.
- c. Name the connection.
- d. Ensure that these default values are set:

Field Value
--- ---
Settings Format Server Info
Connection Type Direct
Database Port 31010

- e. In the **Database Server** field, specify the hostname of your coordinator node.
- f. In the **Database Userid** and **Database Password** fields, specify your authentication credentials:

To use a personal access token that you obtained from Dremio:

In the **Database Userid** field, specify the username for which the PAT was generated.

In the **Database Password** field, paste your personal access token.

To use your Dremio username and password:

In the **Database Userid** field, specify your username.

In the **Database Password** field, your password.

g. (Optional) Click **Ping Server** to test the connection.

h. Click **Finish**.

DbVisualizer creates the connection and opens it.

Was this page helpful?

[Previous

Dremio JDBC Driver (Legacy)](/current/client-applications/drivers/jdbc)[Next

Accelerate Queries](/current/acceleration/)

Supported Versions

Supported Authentication Methods

Prerequisites

Connecting to Dremio

Source:
<https://docs.dremio.com/current/client-applications/microsoft-excel/microsoft-excel-powerpivot>

Version: current [26.x]

On this page

Microsoft Excel PowerPivot

Prerequisites

Ensure that you are using Dremio v22.0 or later.

Ensure that your operating system is 64-bit Windows 10 or later.

Download, install, and configure the [Arrow Flight SQL ODBC driver](#).

If you want to authenticate to Dremio by using a personal access token (PAT), rather than by using a password, generate a PAT. See [Personal Access Tokens](#) for the steps.

Updating the DSN Configuration

Launch ODBC Data Sources on your Windows system.

Select the **System DSN** tab.

Select the DSN entry that you created when you configured the Arrow Flight SQL ODBC driver.

Click **Configure**.

In the **Advanced Properties** section, add the following key/value pair:

Key: quoting

Value: BRACKET

Connecting to Dremio

Open Excel.

Click the **Power Pivot** tab and then click **Manage**.

Select **From Other Sources**.

In the Table Import Wizard, select **Others (OLEDB/ODBC)**.

Click **Next**.

Click **Build**.

In the Data Link Properties dialog, follow these steps:

- a. On the **Provider** tab, select **Microsoft OLE DB Provider for ODBC Drivers**.
- b. Click **Next>>**.
- c. For step 1 on the **Connection** tab, select **Use data source name**, and then select the data source name for the Arrow Flight SQL ODBC driver.
- d. For step 2 on the **Connection** tab, specify the username to use for connections to Dremio, then specify either a password or a personal access token to use with the username.
- e. (Optional) Click **Test Connection** to find out whether the info you specified on this tab is correct.
- f. Click **OK**.

(Optional) Click **Test Connection** to find out whether you can connect to Dremio.

Click **Next**.

Ensure that the option **Select from a list of tables and views to choose the data to import**.

Click **Next**.

Select the tables and views that you want to import data from.

Click **Finish**.

Was this page helpful?

[Previous

Microsoft Excel](/current/client-applications/microsoft-excel/)[Next

Microsoft Power BI](/current/client-applications/microsoft-power-bi/)

Prerequisites

Updating the DSN Configuration

Connecting to Dremio

Source:
<https://docs.dremio.com/current/client-applications/microsoft-power-bi>

Version: current [26.x]

On this page

Microsoft Power BI

Connect Microsoft Power BI to visualize your data and create reports.

You can connect Power BI to Dremio in one of the following ways:

Configure a reusable connection to use in Power BI Desktop, Power BI Gateway, or Power BI Service. Power BI Service can connect to Dremio through DirectQuery or through Power BI Gateway.

Connect to a specific dataset by downloading the `.pbids` file from Dremio and opening it in Power BI Desktop.

Supported Authentication Methods

From Power BI, you can authenticate to Dremio with one of the following methods:

Username and password: Use your Dremio credentials.

Personal access token (PAT): For details, see Personal Access Tokens.

Single sign-on (SSO) through OAuth 2.0: For steps on how to configure SSO, see Enable SSO to Dremio from Power BI.

Connect to Dremio from Power BI

The Power BI connector for Dremio now supports connectivity through the open-source Arrow Database Connectivity (ADBC) driver, which Dremio highly recommends using to connect to Dremio. To enable reports to use ADBC, see Enable Connectivity with ADBC.

Existing connections will continue to work, but we recommend using the embedded ADBC driver for all new reports and migrating existing reports to ADBC to benefit from improved performance and supportability.

To connect to Dremio from Power BI Desktop:

Click **Get data**, search for `Dremio`, select **Dremio Software**, and click **Connect**.

In the Dremio Software dialog, follow these steps:

- a. Use the Flight SQL ADBC driver and in the **Server** field specify your Dremio hostname with the `adbc://` prefix. Example: `adbc://acme-company.dremio.com`
- b. (Optional) Complete the other fields in the dialog as you normally would.
- c. Under **Data Connectivity mode**, select either **Import** or **DirectQuery**. Click **OK**.
- d. For **Authentication Method**, select **Basic** or **Key**.

Basic: Enter your Dremio username and password.

Key: Paste in the personal access token you obtained from Dremio. For details, see [Personal Access Tokens](#).

Click **Connect**.

note

Creating Dataflows through Power BI Service is also supported. To create a dataflow, click **New > Dataflow**. For the data source connection, follow the steps above.

Create a Live Connection to a Dataset from Dremio

You can generate a Microsoft Power BI Data Source (`.pbids`) file that represents a live connection to a dataset that is in Dremio. No actual data is stored in this file, and you can think of it as a shortcut to a Power BI Desktop session with a preconfigured view of your data.

note

The `.pbids` file download option must be enabled for users to have access to this feature. To enable this feature, see [Enable the `.pbids` file download](#).

To create a live connection to a dataset:

In Dremio, navigate to the dataset.

Click the ellipsis (...) next to the dataset name.

Select **Download .pbids file**.

Open the downloaded file in Power BI Desktop.

Authenticate using your preferred method.

Power BI Gateway

To enable Power BI users to connect to Dremio via Power BI Gateway:

Install and configure [Power BI Gateway](#) on a machine that can connect to your Dremio cluster.

In the Power BI Gateway configuration, add Dremio as a data source using the same connection details as above.

Advanced Configuration

Enable Connectivity with ADBC

Dremio supports connectivity through Arrow Database Connectivity (ADBC). To enable this for Power BI Service, see the following options.

Enable the ADBC Option for a New Connection

In Power BI Desktop, click **Get data**.

In the Get Data dialog, locate and select **Dremio Software**, and click **Connect**.

In the Dremio Software dialog, in the **Server** field, specify your hostname with the ``adbc://`` prefix. Example: ``adbc://acme-company.dremio.com``

(Optional) Complete the other fields in the dialog as you normally would.

Click **OK**.

Authenticate using your preferred method, and click **Connect**.

Enable the ADBC Option for an Existing Connection

In Power BI Desktop, go to **Data source settings**, select your source, and click **Change source**.

In the Dremio Software dialog, update the **Server** field by adding the ``adbc://`` prefix before the hostname. Example: ``adbc://acme-company.dremio.com``. If you're unable to edit the source this way, click **Transform data**, then click **Advanced Editor** in the **Home** tab. In the dialog that appears, update the hostname/server with the ``adbc://`` prefix, and click **Done**.

Click **OK**.

Reauthenticate using your preferred method, and click **Connect**.

Enable the `.pbids`` File Download in the Dremio Console

To enable the `.pbids`` file download feature:

In Dremio, go to **Admin > Settings**.

In the **Support** section, enable **Allow downloading of .pbids files**.

Click **Save**.

Enable SSO to Dremio from Power BI

SSO is supported only for datasets that use DirectQuery.

note

SSO only works for reports created using the Dremio Cloud connector in Power BI Desktop. Reports created with the Dremio Software connector cannot use SSO by simply changing credentials, they must be converted first.

To convert existing reports from the Dremio Software connector to the Dremio Cloud connector, you'll need to modify the connection in Power BI's Advanced Editor to change the function from ``Dremio.Databases`` to ``DremioCloud.DatabasesByServerV370``.

Prerequisites: Configure Dremio for Microsoft Entra ID

Before enabling SSO for Power BI reports, Dremio must be configured to use Microsoft Entra ID (Azure AD) as an identity provider.

Required configuration:

Set ``services.coordinator.web.auth.type`` to the Microsoft Entra ID / OIDC provider type configured for your deployment (for example, ``azuread``)

Provide the required OIDC or Azure AD configuration file (``azuread.json`` or equivalent)

Deploy configuration files to all coordinator nodes

Restart Dremio cluster after applying changes

For complete setup steps, see [Configure Microsoft Entra ID](#).

note

All authentication setup must be done through configuration files before enabling SSO in Power BI.

The following steps configure the Power BI side of SSO. These steps assume your Dremio deployment is already configured to authenticate users via Microsoft Entra ID.

To enable SSO for Power BI reports:

Enable SSO for a DirectQuery Report

To enable SSO for a report that uses DirectQuery:

In Power BI Service, open the workspace to which you published the report.

Find the dataset that is associated with the report, click the three dots next to its name, and select **Settings**.

In the settings for the dataset, expand **Data source credentials**.

Click **Edit credentials**.

For **Authentication method**, select **OAuth2**.

In the **Privacy level setting for this data source** field, ensure that **Private** is selected.

Select the check box **Report viewers can only access this data source with their own Power BI identities using DirectQuery**.

Click **Sign in**.

Enable SSO for Reports with Power BI Gateway

To enable SSO when you are using Power BI Gateway:

In Power BI Service, open the workspace to which you published the report.

Find the dataset that is associated with the report, click the three dots next to its name, and select **Settings**.

In the settings for the dataset, expand **Gateway connection**.

Recreate your data source by following these steps:

Select the **Maps to** field.

Select **Manually add to gateway**.

For **Data Source Name**, enter a name for the data source.

For **Data Source Type**, select **Dremio Software**.

For **Server**, enter your Dremio hostname with the ``adbc://`` prefix. Example: ``adbc://acme-company.dremio.com``

For **Authentication Method**, select **OAuth2**.

Click **Add**.

In the **Data source credentials** section, click **Edit credentials**.

For **Authentication method**, select **OAuth2**.

In the **Privacy level setting for this data source** field, ensure that **Private** is selected.

Select the check box **Report viewers can only access this data source with their own Power BI identities using DirectQuery**.

Click **Sign in**.

note

SSO requires the OAuth2 authentication method. Basic authentication and personal access tokens do not support SSO when used through Power BI Gateway.

Arrow Database Connectivity (ADBC) Limitations

ADBC is not enabled by default. It must be enabled by the owner of the report.

NativeQuery is not supported.

Metadata calls are not cached.

Power BI Desktop occasionally caches errors that might affect future connection

attempts until the cache is cleared.

Complex data types such as `MAP` and `INTERVAL` are not supported.

When using DirectQuery, chaining functions is supported, but some complex scenarios may not work as expected. Complex optional parameters for functions are not supported.

Troubleshoot Power BI

Cached Data Issues

If you have previously installed older versions of Power BI Desktop, cached data may interfere with the newer versions of the Flight SQL drivers resulting in connection errors.

Problem

For example, when using Flight SQL ADBC, cached connection data in Power BI could cause the following errors:

```
`ADBC: IOError [] [FlightSQL] [FlightSQL] unresolved address (Unavailable;
GetObjects(GetDBSchemas))`
```

```
`ADBC: IOError [] [FlightSQL] [FlightSQL] connection error: desc = "transport:
authentication handshake failed: credentials: cannot check peer: missing selected ALPN
property. If you upgraded from a grpc-go version earlier than 1.67, your TLS
connections may have stopped working due to ALPN enforcement. For more details,
see: https://github.com/grpc/grpc-go/issues/434" (Unavailable;
GetObjects(GetDBSchemas))`
```

Solution

Clear the Power BI Desktop cache and any cached data source permissions involving Dremio connections by following these steps:

Clear Power BI Desktop Caches.

In Power BI Desktop, go to **File > Options and Settings > Data Source Settings**.

Select **Global Permissions**.

Clear all cached connections by clicking **Clear All Permissions**, or select specific Dremio data sources and click **Clear Permissions**.

After completing these steps, try reconnecting to Dremio using the instructions above.

Large Result Sets

Problem

When fetching data from Dremio with ADBC you may see the following error:

```
`Unexpected error: [FlightSQL] grpc: received message larger than max (43747370 vs. 16777216) (ResourceExhausted; DoGet: endpoint 0: [])`
```

Solution

By default, the ADBC driver accepts only messages up to 16 MiB in size. This can be fixed by updating the Power BI M expression to customize the connection as follows:

```
let
    Source = DremioCloud.DatabasesByServerV370("your-server-here", [
        MaxMessageSize = 67108864 // 64 MiB
    ])
in
    Source
```

Replace `your-server-here` with your actual Dremio server address. The `MaxMessageSize` parameter sets the maximum message size in bytes (67108864 = 64 MiB).

Was this page helpful?

[Previous

Microsoft Excel
PowerPivot](/current/client-applications/microsoft-excel/microsoft-excel-powerpivot)[Ne
xt

Microstrategy Workstation](/current/client-applications/microstrategy)

Supported Authentication Methods

Connect to Dremio from Power BI

Create a Live Connection to a Dataset from Dremio

Power BI Gateway

Advanced Configuration

Enable Connectivity with ADBC

Enable the `.pbids` File Download in the Dremio Console

Enable SSO to Dremio from Power BI

Arrow Database Connectivity (ADBC) Limitations

Troubleshoot Power BI

Cached Data Issues

Large Result Sets

Source:
<https://docs.dremio.com/current/client-applications/drivers/arrow-flight-sql-jdbc-driver>

Version: current [26.x]

On this page

Arrow Flight SQL JDBC

The Arrow Flight SQL JDBC driver is an open-source driver that is based on the specifications for the Java Database Connectivity (JDBC) API. However, the Flight SQL JDBC driver uses Apache Arrow, so it is able to move large amounts of data faster, in part because it does not need to serialize and then deserialize data.

This driver solves a problem that is common to many BI tools that access databases through JDBC. These tools bundle a different JDBC driver for each type of database they support, because each of these databases has their own proprietary driver. Bundling multiple JDBC drivers for multiple databases can be difficult to maintain, and responding to support issues for multiple drivers can be costly. Now, provided that a database has an Apache Arrow Flight SQL endpoint enabled, the JDBC driver can connect to it.

This driver is developed and maintained by the Apache Arrow community. For full technical documentation, see Apache's [Arrow Flight SQL JDBC Driver](#). For Dremio-specific compatibility, version guidance, and release notes, see Dremio's [Arrow Flight SQL JDBC Release Notes](#).

This driver is licensed under [Apache-2.0](#).

tip

Query planning is done on the specified node. To distribute query planning for JDBC connections, configure [secondary coordinator nodes](#) for your deployment.

Prerequisites

One of the following operating systems: Windows, MacOS, or Linux

Supported Java versions: Java 11+

Supported JDK versions: 11, 17, and 21

Requires the following option to be present:

Java 11+ Requirement

```
--add-opens=java.base/java.nio=ALL-UNNAMED
```

Supported Authentication Methods

Use the username and password of the Dremio account that you want to connect with.

Use a username and personal access token (PAT).

Use an OAuth Access Token

Username and Password

Pass a username and password with the ``user`` and ``password`` properties.

Personal Access Tokens Enterprise

Pass a username and personal access token (PAT) with the ``user`` and ``password`` properties. You must URL-encode PATs that you include in JDBC URLs. To encode a PAT locally on your system, you can follow the steps in [URL-encoding Values](#). See [Personal Access Tokens](#) for enabling and creating PATs.

tip

Dremio recommends OAuth access tokens to improve security by reducing the risk of compromised passwords or personal access tokens.

OAuth Access Tokens Enterprise

To create a connection with an OAuth access token, configure the following properties:

``token`` property with the value of the OAuth access token.

``user`` property with the empty string ```` to default to the username included in the access token. If the username is configured in the property value, it must match the username in the access token.

Example Arrow Flight SQL JDBC Connection Using OAuth Access Token

```
import jaydebeapi
jdbc_arrow_flight_url = "jdbc:arrow-flight-sql://{ }:{ }".format("localhost", 32010)
jdbc_arrow_flight_args = { "user": "", "token": dremio_access_token }
jdbc_driver_location_example =
"/Users/me/workspace/drivers/flight-sql-jdbc-driver-18.3.0.jar"
jdbc_arrow_flight_conn =
jaydebeapi.connect("org.apache.arrow.driver.jdbc.ArrowFlightJdbcDriver",
                    jdbc_arrow_flight_url,
                    jdbc_arrow_flight_args,
                    jdbc_driver_location_example)
```

Users can create OAuth access tokens using a local or LDAP username and password, a PAT, or an external JWT. Dremio provides [sample code](#) for each of these cases.

Connecting to Databases

Use this template to create a direct connection to a database that has enabled an Apache Arrow Flight SQL endpoint:

Create direct connection to database

```
jdbc:arrow-flight-sql://<hostname-or-IP-address>:<port-number>/?useEncryption=false[&schema=<optional_schema>][&<properties>]
```

`<optional_schema>`: The name of the schema (datasource or space, including child paths, such as `myDatasource.folder1` and `mySpace.folder1.folder2`) to use by default when a schema is not specified in a query.

`<properties>`: A list of JDBC properties. Values must be URI-encoded.

Use this template to create a direct connection to a Dremio coordinator node:

Create direct connection to Dremio coordinator node

```
jdbc:arrow-flight-sql://<Dremio_coordinator>:32010[/?schema=<optional_schema>][&<properties>]
```

`<Dremio_coordinator>`: The hostname or IP address of the coordinator node in your Dremio cluster.

`<optional_schema>`: The name of the schema (datasource or space, including child paths, such as `myDatasource.folder1` and `mySpace.folder1.folder2`) to use by default when a schema is not specified in a query.

`<properties>`: A list of JDBC properties. Values must be URL-encoded. See URL-encoding Values for suggested steps.

Downloading the Driver

To download the driver, go to [Apache Arrow Flight SQL JDBC](#).

Integrating the driver

To integrate the driver into your development environment, add it to your classpath.

Name of the Class

The name of the class is `org.apache.arrow.driver.jdbc.ArrowFlightJdbcDriver`.

JDBC Properties for Dremio Wire Encryption

If you are setting up encrypted communication between your JDBC client applications and the Dremio server, use the SSL JDBC connection parameters and fully qualified hostname to configure the JDBC connection string and connect to Dremio.

note

This driver does not yet support these features:

Disabling host verification

Impersonation

Properties	Value	Required	Description
---	---	---	---
<code>`useEncryption`</code>	<code>`true`</code> or <code>`false`</code>	[Optional]	If <code>`true`</code> , SSL is enabled. If set to <code>`false`</code> , SSL is not enabled. The default is <code>`true`</code> . If you do not want to use encryption, you must set the value to <code>`false`</code> .
<code>`disableCertificateVerification`</code>	<code>`true`</code> or <code>`false`</code>	[Optional]	If <code>`true`</code> , Dremio does not verify the host certificate against the truststore. The default value is <code>`false`</code> .
<code>`trustStoreType`</code>	string	[Optional]	Default: JKS The trustStore type. Allowed values are : <code>`JKS`</code> , <code>`PKCS12`</code> If the <code>useSystemTrustStore</code> option is set to true (on Windows only), the allowed values are: <code>`Windows-MY`</code> , <code>`Windows-ROOT`</code> Import the certificate into the Trusted Root Certificate Authorities and set <code>`trustStoreType=Windows-ROOT`</code> . Also import the certificate into Trusted Root Certificate Authorities or Personal and set <code>`trustStoreType=Windows-MY`</code> .
<code>`trustStore`</code>	string	[Optional]	Path to the truststore. If not provided, the default Java truststore is used (usually <code>`\$JAVA_HOME/lib/security/cacerts`</code>) and the <code>trustStorePassword</code> parameter is ignored.
<code>`useSystemTrustStore`</code>	<code>`true`</code> or <code>`false`</code>	[Optional]	By default, the value is <code>`true`</code> . Bypasses <code>trustStoreType</code> and automatically picks the correct Truststore based on the operating system: Keychain on MacOS, <u>Local Machine and Current User Certificate Stores</u> on Windows, and default truststore on other operating systems. If you are using an operating system other than MacOS or Windows, you must use the <code>`trustStorePassword`</code> property to pass the password of the truststore. Here is an example of a connection string for Linux: <code>`jdbc:arrow-flight-sql://localhost:32010?trustStorePassword=Pc0_IL'Opjn\$vSDcv:%Q0@@buc`</code>
<code>`trustStorePassword`</code>	string	[Optional]	Password to the truststore.

Parameterized Queries with Prepared Statements

Prepared statements allow you to dynamically pass parameters to SQL queries using placeholders, ensuring safer query execution by separating the query structure from the values in parameters. With a prepared statement, you can set parameters (``?``) at runtime using set methods to reuse queries with different values.

note

This feature requires Apache Arrow 18.3.0 or later. It supports ``SELECT`` statements and ``DML`` statements.

To use parameterized queries with prepared statements, follow these steps:

Use the ``prepareStatement()`` method to define a query with parameters, which act as placeholders for dynamic values.

Set the values by replacing each parameter with a value using the appropriate set methods.

Ensure all parameters are set before running the query, with indexing starting at 1. If parameters are not set before running the query, JDBC throws an exception.

Call ``executeQuery()`` to run the `SELECT` query and retrieve results, or

``executeUpdate()`` to run the DML query and retrieve the count of modified records.

Java examples for SELECT and DML queries

```
PreparedStatement preparedStatement = connection.prepareStatement(
    "SELECT * FROM employees WHERE department = ? AND salary > ?");
preparedStatement.setString(1, "Engineering");
preparedStatement.setDouble(2, 75000);
ResultSet resultSet = preparedStatement.executeQuery();
```

```
PreparedStatement preparedStatement = connection.prepareStatement(
    "DELETE FROM employees WHERE department = ? AND salary > ?");
preparedStatement.setString(1, "Engineering");
preparedStatement.setDouble(2, 75000);
int rowsUpdated = preparedStatement.executeUpdate();
```

Supported Data Types and Set Methods

Column Data Type	Supported Set Methods
Integer	<code>`setInt()`</code> , <code>`setShort()`</code> , <code>`setNull()`</code>
Numeric	<code>`setInt()`</code> , <code>`setShort()`</code> , <code>`setLong()`</code> , <code>`setBigDecimal()`</code> , <code>`setNull()`</code>
Decimal	<code>`setShort()`</code> , <code>`setInt()`</code> , <code>`setLong()`</code> , <code>`setBigDecimal()`</code> , <code>`setNull()`</code>
BigInt	<code>`setShort()`</code> , <code>`setInt()`</code> , <code>`setLong()`</code> , <code>`setBigDecimal()`</code> , <code>`setNull()`</code>
Double	<code>`setDouble()`</code> , <code>`setFloat()`</code> , <code>`setNull()`</code>
Float	<code>`setFloat()`</code> , <code>`setNull()`</code>
Char	<code>`setString()`</code> , <code>`setNull()`</code>
Varchar	<code>`setString()`</code> , <code>`setNull()`</code>
Boolean	<code>`setBoolean()`</code> , <code>`setNull()`</code>
Time	<code>`setTime()`</code> , <code>`setNull()`</code>
Timestamp	<code>`setTimestamp()`</code> , <code>`setNull()`</code>
Date	<code>`setNull()`</code>
VarBinary	<code>`setBytes()`</code> , <code>`setNull()`</code>

Limitations

The JDBC client does not support the ``setDate()`` method due to mismatched date encoding formats between the Arrow Flight JDBC client and Dremio.

Managing Workloads

Dremio administrators can use the Arrow Flight server endpoint to manage query workloads by adding the following properties to connections created by Flight clients:

Flight Client Property	Description
<code>`ROUTING_ENGINE`</code>	Name of the engine to use to process all queries issued during the current session.
<code>`ROUTING_QUEUE`</code>	Name of the workload management queue. Used only during authentication.

| ``ROUTING_TAG`` | Tag name associated with all queries executed within a Flight session. Used only during authentication. |
| ``SCHEMA`` | Default schema path to the dataset that the user wants to query. |

URL-encoding Values

To encode a personal access token (PAT) or property value locally on your system, you can follow these steps:

In a browser window, right-click an empty area of the page and select **Inspect**.

Click **Console**.

Type ``encodeURIComponent("<PAT-or-value>")``, where ``<PAT-or-value>`` is the personal access token that you obtained from Dremio or the value of a supported JDBC property. The URL-encoded PAT or value appears on the next line. You can highlight it and copy it to your clipboard.

Differences between the Arrow Flight SQL JDBC Driver and the Legacy Dremio JDBC Driver

The Arrow Flight SQL JDBC driver differs from the Dremio JDBC driver (legacy) in the following:

Requires Java 11+.

Supports ``ResultSet.getBoolean()`` on ``varchar`` columns in which boolean values are represented as these strings: "0", "1", "true", "false".

Supports null Calendar in calls to ``ResultSet.getDate()``, ``ResultSet.getTime()``, and ``ResultSet.getTimestamp()``

When a call to one of these methods has no ``Calendar`` parameter, or the ``Calendar`` parameter is ``null``, the Flight JDBC driver uses the default timezone when it constructs the returned object.

Supports ``ResultSet.getDate()``, ``ResultSet.getTime()``, and ``ResultSet.getTimestamp()`` on ``varchar`` columns in which dates, times, or timestamps are represented as strings.

Supports `varchar` values that represents numeric values in calls to ``ResultSet.getInteger()``, ``ResultSet.getFloat()``, ``ResultSet.getDouble()``, ``ResultSet.getShort()``, ``ResultSet.getLong()``, and ``ResultSet.getBigDecimal()``

Supports integer values in calls to ``getFloat()``
Integers returned gain one decimal place.

Supports the native SQL complex types ``List``, ``Map``, and ``Struct``
The Dremio JDBC driver (legacy) uses String representations of these types.

Supports using the Interval data type in SQL functions.

Removes support for calling ``ResultSet.getBinaryStream()`` on non-binary data types. Though such support exists in traditional JDBC drivers, it is not in the specification for the JDBC API.

note

Calling ``DatabaseMetadata.getCatalog()`` when connected to Dremio returns empty. Other ``DatabaseMetadata`` methods return null values in the ``TABLE_CAT`` column. This is expected behavior because Dremio does not have a catalog.

Limitations

Impersonation is not supported.

Supported Conversions from Dremio Datatypes to JDBC Datatypes

DREMIO TYPE	JDBC	ARROW TYPE
---	---	---
BIGINT	Int	
BIT	Bool	
DATE	Date	
DECIMAL	Decimal	
DOUBLE	FloatingPoint(DOUBLE)	
FIXEDSIZEBINARY	FixedSizeBinary	
FLOAT	FloatingPoint(SINGLE)	
INT	Int	
INTERVAL_DAY_SECONDS	Interval(DAY_TIME)	
INTERVAL_YEAR_MONTHS	Interval(YEAR_MONTH)	
LIST	List	
MAP	Map	
NULL	Null	
OBJECT	Not Supported	
STRUCT	Struct	
TIME	Time(MILLISECOND)	
TIMESTAMP	Timestamp(MILLISECOND)	
VARBINARY	Binary	
VARCHAR	Utf8	

Was this page helpful?

[\[Previous](#)

[Drivers\]\(/current/client-applications/drivers/\)\[Next](#)

[Arrow Flight SQL ODBC\]\(/current/client-applications/drivers/arrow-flight-sql-odbc-driver\)](#)

[Prerequisites](#)

[Supported Authentication Methods](#)

[Username and Password](#)

[Personal Access Tokens Enterprise](#)

[OAuth Access Tokens Enterprise](#)

[Connecting to Databases](#)

[Downloading the Driver](#)

Integrating the driver

Name of the Class

JDBC Properties for Dremio Wire Encryption

Parameterized Queries with Prepared Statements

Supported Data Types and Set Methods

Limitations

Managing Workloads

URL-encoding Values

Differences between the Arrow Flight SQL JDBC Driver and the Legacy Dremio JDBC Driver

Limitations

Supported Conversions from Dremio Datatypes to JDBC Datatypes

Source:
<https://docs.dremio.com/current/client-applications/drivers/jdbc>

Version: current [26.x]

On this page

Dremio JDBC Driver (Legacy)

note

The [Arrow Flight SQL JDBC driver](#) is recommended for use for connectivity to Dremio. If you are using a client application that bundles or is certified with the Dremio JDBC driver, you can continue to use this driver. The Dremio JDBC driver will be not enhanced or fixed with the exception of critical security fixes.

The legacy [Dremio JDBC driver](#) is included as a part of Dremio installations under ``<DREMIO_HOME>/jars/jdbc-driver/``. The main JAR Class is ``com.dremio.jdbc.Driver``. You can also download the JDBC driver from [here](#). This driver is licensed under [Apache-2.0](#).

A new version of the JDBC driver is made available with every release of Dremio software. However, this doesn't mean changes or new features were introduced in a driver release. Only when actual changes are made to a driver will the [JDBC driver release notes](#) be published.

Prerequisites

As of Dremio JDBC driver 25.0+, supported Java versions: Java 11+

-

Supported JDK versions: 11, 17, and 21

Requires the following option to be present:

Java 11+ Requirement

```
--add-opens=java.base/java.nio=ALL-UNNAMED
```

Supported Authentication Methods

Username and Password

Use the username and password of the Dremio account that you want to connect with.

Personal Access Tokens Enterprise

Use a username and personal access token (PAT). To generate a PAT, see [Creating a PAT](#).

tip

Dremio recommends OAuth access tokens to improve security by reducing the risk of compromised passwords or personal access tokens.

OAuth Access Tokens Enterprise

To create a connection with an OAuth access token, configure the following properties:

``token_type`` with a value of ``access_token``

``password`` with the value of the access token

``user`` with an empty string ```` to default to the username included in the access token. If the username is configured in the property value, it must match the username in the access token.

Example Legacy JDBC Connection Using Dremio Access Token

```
import jaydebeapi
jdbc_url = "jdbc:dremio:direct={}:{}".format("localhost", 31010)
jdbc_args = {"user": "", "password": dremio_access_token, "token_type": "access_token"}

jdbc_driver_location_example =
"/Users/me/workspace/drivers/dremio-jdbc-driver-25.3.0-SNAPSHOT.jar"
jdbc_conn = jaydebeapi.connect("com.dremio.jdbc.Driver",
                               jdbc_url,
                               jdbc_args,
                               jdbc_driver_location_example)
```

Users can create OAuth access tokens using a local or LDAP username and password, a

PAT, or an external JWT. Dremio provides [sample code](#) for each of these cases.

External JWT

To use an external JWT directly from an [external token provider](#), configure the following properties:

``token_type`` with a value of ``jwt``

``password`` with the value of the external JWT

``user`` with the empty string ```` to default to the username included in the external JWT. If the username is configured in the property value, it must match the username in the external JWT.

Dremio provides [sample code](#) for requesting an external JWT from Microsoft Entra ID.

tip

Dremio recommends OAuth access tokens obtained through token exchange over an external JWT. The Dremio OAuth access token is typically smaller than an external JWT and verification is faster.

Setup

You can set up the JDBC driver in the following manner:

Connect directly to the Dremio server

Connect to the Dremio server via Zookeeper

Tip: To distribute query planning for JDBC connections, configure [secondary coordinator nodes](#) for your deployment.

Connecting directly to Dremio

The following configuration establishes a direct connection to a Dremio coordinator node.

Planning is done on the specified node.

Connect directly to Dremio coordinator node

```
jdbc:dremio:direct=<DREMIO_COORDINATOR>:31010[;schema=<OPTIONAL_SCHEMA>]
```

Connecting to ZooKeeper

The following configuration establishes a distributed connection to Dremio coordinator nodes through a

Zookeeper quorum. Planning is distributed across the available coordinator nodes.

Connect to Dremio coordinator node with ZooKeeper

```
jdbc:dremio:zk=<ZOOKEEPER_QUORUM>:2181[;schema=<OPTIONAL_SCHEMA>]
```

Multiple Dremio Clusters in the same ZooKeeper Quorum

Cluster A

```
jdbc:dremio:zk=<ZOOKEEPER_QUORUM>:2181/path/to/ClusterA
```

Cluster B

```
jdbc:dremio:zk=<ZOOKEEPER_QUORUM>:2181/path/to/ClusterB
```

Construct a Prepared Statement with Dynamic Parameters

Dremio supports using parameters in prepared statements for SELECT queries.

The parameter marker is `?` in prepared statements. To execute a prepared statement, you must set the parameter marker with one of the supported set methods.

The example below uses the Date type parameter and the `setDate` set method. For set methods, the first argument is the index of the parameter marker in the SQL query, starting from 1. The second argument is the value for the parameter marker. After you set the parameter, you can execute the prepared statement by calling the `executeQuery()` method on the prepared statement.

Example prepared statement with parameters

```
public class HelloWorld {
    public static void main(String[] args) {
        try (PreparedStatement stmt = getConnection().prepareStatement("SELECT * FROM
(values (DATE '2024-02-20'), (null)) AS a(id) WHERE id=?")) {
            Date date = Date.valueOf(LocalDate.of(2024, 02, 20));
            stmt.setDate(1, date);
            try (ResultSet rs = stmt.executeQuery()) {
                assertThat(rs.getMetaData().getColumnCount()).isEqualTo(1);
                assertThat(rs.next()).isTrue();
                assertThat(rs.getDate(1)).isEqualTo(date);
                assertThat(rs.next()).isFalse();
            }
        }
    }
}
```

The example below demonstrates how to reuse the same prepared statement by defining a different set method and parameter value.

Example prepared statement with different set method and parameters

```
public class HelloWorld {
    public static void main(String[] args) {
        try (PreparedStatement stmt = getConnection().prepareStatement("SELECT * FROM
(values (DATE '2024-02-20'), (null)) AS a(id) WHERE id=?")) {
            Date date = Date.valueOf(LocalDate.of(2024, 02, 20));
```

```

stmt.setDate(1, date);
try (ResultSet rs = stmt.executeQuery()) {
    assertThat(rs.getMetaData().getColumnCount()).isEqualTo(1);
    assertThat(rs.next()).isTrue();
    assertThat(rs.getDate(1)).isEqualTo(date);
    assertThat(rs.next()).isFalse();
}
stmt.setDate(1, Date.valueOf(LocalDate.of(2025, 02, 20)));
try (ResultSet rs = stmt.executeQuery()) {
    assertThat(rs.next()).isFalse();
}
}
}
}
}

```

The following example shows how to use more than one parameter in a prepared statement.

Example prepared statement with two parameters

```

public class HelloWorld {
    public static void main(String[] args) {
        try (PreparedStatement stmt = getConnection().prepareStatement("SELECT * FROM
(values (1), (2), (null)) AS a(id) WHERE id = ? OR id < ?")) {
            stmt.setInt(1, 1);
            stmt.setInt(2, 3);
            try (ResultSet rs = stmt.executeQuery()) {
                assertThat(rs.getMetaData().getColumnCount()).isEqualTo(1);
                assertThat(rs.next()).isTrue();
                assertThat(rs.getInt(1)).isEqualTo(1);
                assertThat(rs.next()).isFalse();
            }
        }
    }
}

```

Set Methods for Prepared Statements with Parameters

To execute a prepared statement, you must set the parameter marker with one of the supported set methods listed in the table below.

Column Data Type	Supported Set Methods
---	---
Integer	setInt(), setShort(), setNull()
Numeric	setInt(), setShort(), setLong(), setBigDecimal(), setNull()
Decimal	setShort(), setInt(), setLong(), setBigDecimal(), setNull()
BigInt	setShort(), setInt(), setLong(), setBigDecimal(), setNull()
Double	setDouble(), setFloat(), setNull()
Float	setFloat(), setNull()
Char	setString(), setNull()
Varchar	setString(), setNull()

Boolean	setBoolean(), setNull()	
Time	setTime(), setNull()	
Timestamp	setTimestamp(), setNull()	
Date	setDate(), setNull()	
VarBinary	setNull(), setBytes()	

JDBC Parameters for Dremio Wire Encryption

If you are setting up encrypted communication between your JDBC client applications and the Dremio server, use the SSL JDBC connection parameters and a fully qualified host name to configure the JDBC connection string and connect to Dremio:

Parameter	Value	Required	Description
---	---	---	---
ssl	true/false	[Optional]	If true, SSL is enabled. If not set or set to false, SSL is not enabled.
trustStoreType	string	[Optional]	Default: JKS The trustStore type. Allowed values are : JKS PKCS12 If the useSystemTrustStore option is set to true (on Windows only), the allowed values are: Windows-MY Windows-ROOT Import the certificate into the Trusted Root Certificate Authorities and set `trustStoreType=Windows-ROOT`. Also import the certificate into Trusted Root Certificate Authorities or Personal and set `trustStoreType=Windows-MY`.
trustStore	string	[Optional]	Path to the truststore. If not provided, the default Java truststore is used (usually \$JAVA_HOME/lib/security/cacerts) and the trustStorePassword parameter is ignored.
useSystemTrustStore	true/false	[Optional]	By default, the value is `true`. Bypasses trustStoreType and automatically picks the correct Truststore based on the operating system: Keychain on MacOS, <u>Local Machine and Current User Certificate Stores</u> on Windows, and default truststore on other systems.
trustStorePassword	string	[Optional]	Password to the truststore.
disableHostVerification	true/false	[Optional]	If true, Dremio does not verify that the host in the certificate is the host we are connecting to. False by default. (Hostname verification follows the specification in RFC2818)
disableCertificateVerification	true/false	[Optional]	If true, Dremio does not verify the host certificate against the truststore. False by default.

Optional Advanced JDBC Driver Properties

Parameter	Value	Description
---	---	---
impersonation_target	string	When inbound impersonation is configured, `impersonation_target` is used for authorization, so it must have permission to the queried datasets, and `impersonation_target` appears as the identity that submitted the queries. The username used to establish the connection must be mapped to `impersonation_target` in the impersonation policy for the Dremio service, otherwise, the connection fails with an authorization error. In the policy, the user used to establish the connection is the `proxy_principle` and `impersonation_target` is its `target_principle`. For more information on configuring policies, see <u>Inbound Impersonation</u> .
routing_queue	string	Specifies the queue to use for processing queries while a connection is open. For more information, see <u>Query Tagging & Direct Routing Configuration</u> .

| routing_tag | string | Sets a tag for rule processing. The specified tag is associated with all queries executed while a connection is open. Rules can check for the presence of a tag with the function `tag()`. For more information, see [Query Tagging & Direct Routing Configuration](#). |

| token_type | string | The type of the token in the `password` field. Valid values are `jwt` for [external tokens](#), `access_token` for OAuth access token, or `personal_access_token` for [personal access tokens](#). If you are using your Dremio password, omit the `token_type` property. |

SOCKS Proxy Connection Parameters

If you want to connect to Dremio Cloud through a SOCKS proxy, use these connection parameters:

Parameter	Type	Description	Default Value	Required?
socksProxyHost	string	The IP address or hostname of the SOCKS proxy.	N/A	Yes
socksProxyPort	integer	The port to use on the SOCKS proxy.	1080	No
socksProxyUsername	string	The username to use for connections.	N/A	No
socksProxyPassword	string	The password to use for connections.	N/A	Only if a username is specified.

Was this page helpful?

[\[Previous](#)

[Arrow Flight SQL ODBC\]\(/current/client-applications/drivers/arrow-flight-sql-odbc-driver\)](#)

[Prerequisites](#)

[Supported Authentication Methods](#)

[Username and Password](#)

[Personal Access Tokens Enterprise](#)

[OAuth Access Tokens Enterprise](#)

[External JWT](#)

[Setup](#)

[Construct a Prepared Statement with Dynamic Parameters](#)

[Set Methods for Prepared Statements with Parameters](#)

[JDBC Parameters for Dremio Wire Encryption](#)

[Optional Advanced JDBC Driver Properties](#)

[SOCKS Proxy Connection Parameters](#)

Source:
<https://docs.dremio.com/current/client-applications/drivers/arrow-flight-sql-odbc-driver>

Version: current [26.x]

On this page

Arrow Flight SQL ODBC

Starting with Dremio v22.0, you can use the Arrow Flight SQL ODBC driver to connect to Dremio from ODBC client applications. This driver is licensed under [GNU Library General Public License, Version 2](#).

Supported Operating Systems

You can use the driver on systems that run the following 64-bit operating systems:

Linux: RedHat/CentOS

Windows 10 and later

macOS (Intel processors only)

Apple Silicon Compatibility

The Arrow Flight SQL ODBC driver is not supported on Apple Silicon M1, M2, and M3 processors. While previous workarounds using Rosetta may have been available, they are no longer reliable and may not work with current versions. For Apple Silicon Mac computers, consider using alternative connection methods such as the [Arrow Flight SQL JDBC driver](#).

Authentication Methods

Dremio supports several authentication methods for client connections.

Username and Password

Pass a username and password with the **UID** and **PWD** properties.

Personal Access Tokens Enterprise

Pass a username and personal access token (PAT) with the **UID** and **PWD** properties, respectively. See [Personal Access Tokens](#) for enabling and creating PATs.

tip

Dremio recommends OAuth access tokens to improve security by reducing the risk of compromised passwords or personal access tokens.

OAuth Access Tokens Enterprise

To create a connection with an OAuth access token, configure the **TOKEN** property with the value of the OAuth access token.

Example Arrow Flight SQL ODBC Connection using OAuth Access Tokens

```
import pyodbc

with pyodbc.connect(
    # Default location on Linux
    Driver='/opt/arrow-flight-sql-odbc-driver/lib64/libarrow-odbc.so.0.9.1.168',
    HOST='my.odbc.host',
    PORT=32010,
    useEncryption='true',
    TOKEN=dremio_access_token,
    autocommit=True,
) as conn:
    with conn.cursor() as cursor:
        cursor.execute('select * from test_table')
        results = cursor.fetchall()
```

Users can create OAuth access tokens using a local or LDAP username and password, a PAT, or an external JWT. Dremio provides [sample code](#) for each of these cases.

Downloading and Installing

Windows

Linux

macOS

Downloading and Installing on Windows

note

The Arrow Flight SQL ODBC driver is not available for 32-bit Windows versions.

note

If you plan to use Microsoft Power BI Desktop April 2022 or later to connect to Dremio, you do not need to use this driver. Power BI Desktop April 2022 and later includes a connector that you can use to connect to Dremio. See [Connecting from Microsoft Power BI](#).

To download and install the Arrow Flight SQL ODBC driver:

Download the Windows 64-bit version of the driver from the [ODBC driver download page](#).

Run the installer.

(Optional) In the **User Account Control** page, click **Yes**. This page appears only if there is user account control configured on your Windows machine.

In the **Welcome to Dremio** page, click **Next**.

Click **Install**.

In the **Installation Complete** page, click **Next**.

In the **Completing Arrow Flight SQL ODBC Driver Setup Wizard** page, click **Finish**.

Next, configure the driver.

Downloading and Installing on Linux

To download and install the Arrow Flight SQL ODBC driver:

Download the Linux version of the driver from the [ODBC driver download page](#).

Run the following command to install the driver and automatically create the data source name (DSN) `Arrow Flight SQL ODBC DSN`:

Install driver and create data source name (DSN)

```
sudo yum localinstall <dremio-odbc-rpm-path>
```

Next, configure the driver.

Downloading and Installing on macOS

To download and install the Arrow Flight SQL ODBC driver:

Intel Macs Only

This driver only supports Intel-based Macs. It is not compatible with Apple Silicon M1, M2, and M3 processors.

Download the macOS version of the driver from the [ODBC driver download page](#).

Go to the download location and double-click the downloaded `.dmg` file.

Double-click the `.pkg` file.

In the **Welcome to the ODBC Driver for Arrow Flight SQL Installer** page, click **Continue**.

In the **Standard Install on "Macintosh HD"** page, Click **Install**. Optionally, if you want to change the install location, click **Change Install Location** and navigate to the new location.

In the **Installer is trying to install new software** dialog, specify your macOS password. Then, click **Install Software**.

After the installation is complete, click **Close**.

Next, configure the driver.

Configuring

Configuring on Windows

To configure the System DSN:

danger

Do not follow these steps if you are using Microsoft Power BI Desktop to connect to Dremio. For the steps for configuring Power BI, see [Connecting from Microsoft Power BI](#).

note

If you want to use a personal access token (PAT), rather than a password, for authenticating to Dremio, generate a PAT. See [Personal Access Tokens](#) for the steps.

Go to **Start Menu > Window Administrative Tools**. Click **ODBC Data Sources (64-bit)**.

In the **ODBC Data Source Administrator (64-bit)** dialog, click **System DSN**.

Select **Arrow Flight SQL ODBC DSN** and click **Configure**.

In the **HOST** field, specify the hostname of the server or its IP address.

In the **PORT** field, specify the port to use for connections from Arrow Flight SQL ODBC client applications, which is 32010 by default.

Specify client information in the appropriate fields for your authentication type:

	Field	Username and Password	Personal Access Token	OAuth Access Token
	---	---	---	---
	UID	Username	Username	Do not specify
	PWD	Password	Personal access token	Do not specify
	TOKEN	Do not specify	Do not specify	OAuth access token

In the **UseEncryption** field, specify one of these values:

`true`, if Dremio is configured for encrypted communication with your Arrow Flight SQL ODBC client applications.

`false`, if Dremio is not configured for encrypted communication with your Arrow Flight SQL ODBC client applications. Dremio is unencrypted by default.

For additional parameters, see Connection Parameters.

If you ever need to enable tracing for troubleshooting problems with the driver, click the **Tracing** tab in the **ODBC Data Source Administrator (64-bit)** dialog, set the log-file path, and then click **Start Tracing Now**.

Configuring on Linux

note

Before configuring, ensure that unixODBC is installed.

If you want to base your configuration on examples, copy the content of the `odbc.ini` and `odbcinst.ini` files in the `/opt/arrow-flight-sql-odbc-driver/conf` directory and

paste the content into your system ``/etc/odbc.ini`` and ``/etc/odbcinst.ini`` files.

To configure the properties in the `odbc.ini` file:

In the **HOST** field, specify the hostname of the server or its IP address.

In the **PORT** field, specify the port to use for connections from Arrow Flight SQL ODBC client applications, which is 32010 by default.

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
---	---	---	---
UID	Username	Username	Do not specify
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

In the **UseEncryption** field, specify one of these values:

``true``, if Dremio is configured for encrypted communication with your Arrow Flight SQL ODBC client applications.

``false``, if Dremio is not configured for encrypted communication with your Arrow Flight SQL ODBC client applications. Dremio is unencrypted by default.

For additional parameters, see Connection Parameters.

note

To find out unixODBC has created your ``odbc.ini`` and ``odbcinst.ini`` files, run this command:

```
odbcinst -j
```

If you ever need to enable tracing for troubleshooting problems with the driver, see the help for unixODBC.

Configuring on macOS

note

Before configuring, ensure that ODBC Manager is installed.

Launch ODBC Manager.

On the System DSN page, select **Arrow Flight SQL ODBC DSN** and click **Configure**.

(Optional) Change the DSN.

In the **Host** field, specify the hostname of the server or its IP address.

In the **Port** field, specify the port to use for connections from Arrow Flight SQL ODBC client applications, which is 32010 by default.

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
---	---	---	---
UID	Username	Username	Do not specify
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

In the **UseEncryption** field, specify one of these values:

``true``, if Dremio is configured for encrypted communication with your Arrow Flight SQL ODBC client applications.

``false``, if Dremio is not configured for encrypted communication with your Arrow Flight SQL ODBC client applications. Dremio is unencrypted by default.

For additional parameters, see Connection Parameters.

If you ever need to enable tracing for troubleshooting problems with the driver, see the help for your driver manager.

Connection Parameters

Primary Connection Parameters

Use these parameters to configure basic connection details such as what data source to connect with.

note

The Arrow Flight SQL ODBC driver does not support password-protected ``.pem`` / ``.crt`` files or multiple ``.crt`` certificates in a single ``.pem`` / ``.crt`` file.

Name	Type	Description	Default Value
---	---	---	---
Host	string	Sets the IP address or hostname for the Dremio server. If you specify an IP address and you set the TLS connection parameter <code>`useEncryption`</code> to <code>`true`</code> , ensure that the <code> `/etc/hosts/`</code> file includes an entry to map the IP address to the host.	None
Port	integer	Sets the TCP port number that Dremio uses to listen to connections from Arrow Flight SQL ODBC clients.	32010
Schema	string	Provides the name of the database schema to use by default when a schema is not specified in a query. However, this does not prevent queries from being issued for other schemas. Such queries must explicitly include the schema.	None

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
---	---	---	---
UID	Username	Username	Do not specify
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

TLS Connection Parameters

Use the following parameters to configure TLS encryption and verification methods for

regular connections.

Name	Type	Description	Default Value
useEncryption	integer	Configures the client to use a TLS-encrypted connection to communicate with the Dremio server. Accepted values: * `true`, the client communicates with the Dremio server only using TLS encryption. This is the default value. Therefore, communication between the client application and your Dremio server must be encrypted if you do not override this default value. See the configuration of Arrow Flight TLS for Dremio on Kubernetes or Dremio standalone clusters for more information. * `false`, TLS encryption is disabled with the client. If you specify this value, ensure that the encryption of communication between the client application and your Dremio server is not configured.	true
disableCertificateVerification	integer	Specifies whether the driver should verify the host certificate against the trust store. Accepted values: * `false`, the driver verifies the certificate against the trust store. * `true`, the driver does not verify the certificate against the trust store.	false
useSystemTrustStore	integer	Controls whether to use a CA certificate from the system's trust store, or from a specified `.pem` file. Accepted values: * `true`, the driver verifies the connection using a certificate in the system trust store. * `false`, the driver verifies the connection using the `.pem` file specified by the `trustedCerts` parameter.	`true` on Windows and macOS, `false` on Linux (which does not have a system truststore)
trustedCerts	string	The full path of the `.pem` file containing certificates trusted by a CA, for the purpose of verifying the server. If this option is not set, the driver defaults to using the trusted CA certificates `.pem` file installed by the driver. The exact file path varies according to the operating system on which the driver is installed. The path for the Windows driver differs from the path set for the macOS driver. The TLS connection fails if you do not specify a value when `useEncryption` is `true` and `disableCertificateVerification` is `false`.	N/A
hideSQLTablesListing	boolean	Prevents Microsoft Excel 16.95+ from crashing by hiding the list of available sources in Microsoft Excel's Query Dialog. Set to `true` to enable. Only for Intel-based Mac computers.	`false`

Advanced Parameters

Name	Type	Description	Default Value
quoting	string	Specifies which type of character to use to delimit values in queries. The value can be BACK_TICK, BRACKET, or DOUBLE_QUOTE.	DOUBLE_QUOTE
routing_queue	string	Specifies the queue to route queries to during a session. Direct Routing is used to specify the exact queue and execution cluster to run queries on for a given ODBC session. With Direct Routing, workload-management (WLM) rules are not considered; instead, queries are routed directly to the specified queue. For more information, see Workload Management .	N/A
routing_tag	string	When this parameter is set, the specified tag is associated with all queries executed within a session. Rules can check for the presence of a tag with the function "tag()". For more information, see Workload Management .	N/A
stringColumnLength	string	The maximum length of data in columns of the STRING datatype and of complex datatypes. The range is 1 to 2147483647.	1024

Logging

You can inspect and extract client-side driver logs through the macOS logging infrastructure, because every macOS installation comes with an embedded Console application that allows you to filter by log types. This feature is only available for Intel-based Mac computers, and the log activity of the ODBC driver mostly consists of ODBC API calls.

To start logging:

Open the Console application.

In the search box, select **Excel** in the **PROCESS** dropdown, and **odbc** in the **ANY** dropdown.

!Filter by Excel and ODBC for logs in the Console application.

Click **Start**.

Supported Conversions from Dremio Datatypes to ODBC Datatypes

Dremio Data Types	SQL\C_BINARY	SQL\C_BIT	SQL\C_CHAR	SQL\C_WCHAR	SQL\C_STINYINT	SQL\C_UTINYINT	SQL\C_SSHORT	SQL\C_USHORT	SQL\C_SLONG	SQL\C_ULONG	SQL\C_SBIGINT	SQL\C_UBIGINT	SQL\C_FLOAT	SQL\C_DOUBLE	SQL\C_NUMERIC	SQL\C_DATE	SQL\C_TIME	SQL\C_TIMESTAMP	SQL\C_GUID	SQL\C_INTERVAL*	
BOOLEAN	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
VARBINARY	Y	Invalid	N	N	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
DATE	N	Invalid	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
FLOAT	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
DECIMAL	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Invalid	Invalid	Invalid
DOUBLE	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
INTERVAL (day to seconds)	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	Invalid	Invalid	Invalid
INTERVAL (years to months)	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	Invalid	Invalid	Invalid
INT	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
BIGINT	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
TIME	N	N	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
TIMESTAMP	N	N	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
VARCHAR	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid
STRUCT	N	N	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
LIST	N	Invalid	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Invalid | Invalid | Invalid | Invalid | Invalid | Invalid | Invalid | Invalid | Invalid |

Was this page helpful?

[Previous

Arrow

Flight

SQL

JDBC](/current/client-applications/drivers/arrow-flight-sql-jdbc-driver)[Next

Dremio JDBC Driver (Legacy)](/current/client-applications/drivers/jdbc)

Supported Operating Systems

Authentication Methods

Username and Password

Personal Access Tokens Enterprise

OAuth Access Tokens Enterprise

Downloading and Installing

Downloading and Installing on Windows

Downloading and Installing on Linux

Downloading and Installing on macOS

Configuring

Configuring on Windows

Configuring on Linux

Configuring on macOS

Connection Parameters

Primary Connection Parameters

TLS Connection Parameters

Advanced Parameters

Logging

Supported Conversions from Dremio Datatypes to ODBC Datatypes

Source:
<https://docs.dremio.com/current/client-applications/drivers/arrow-flight-sql-odbc-driver/>

Version: current [26.x]

On this page

Arrow Flight SQL ODBC

Starting with Dremio v22.0, you can use the Arrow Flight SQL ODBC driver to connect to Dremio from ODBC client applications. This driver is licensed under [GNU Library General Public License, Version 2](#).

Supported Operating Systems

You can use the driver on systems that run the following 64-bit operating systems:

Linux: RedHat/CentOS

Windows 10 and later

macOS (Intel processors only)

Apple Silicon Compatibility

The Arrow Flight SQL ODBC driver is not supported on Apple Silicon M1, M2, and M3 processors. While previous workarounds using Rosetta may have been available, they are no longer reliable and may not work with current versions. For Apple Silicon Mac computers, consider using alternative connection methods such as the [Arrow Flight SQL JDBC driver](#).

Authentication Methods

Dremio supports several authentication methods for client connections.

Username and Password

Pass a username and password with the **UID** and **PWD** properties.

Personal Access Tokens Enterprise

Pass a username and personal access token (PAT) with the **UID** and **PWD** properties, respectively. See [Personal Access Tokens](#) for enabling and creating PATs.

tip

Dremio recommends OAuth access tokens to improve security by reducing the risk of compromised passwords or personal access tokens.

OAuth Access Tokens Enterprise

To create a connection with an OAuth access token, configure the **TOKEN** property with the value of the OAuth access token.

Example Arrow Flight SQL ODBC Connection using OAuth Access Tokens

```
import pyodbc

with pyodbc.connect(
    # Default location on Linux
    Driver='/opt/arrow-flight-sql-odbc-driver/lib64/libarrow-odbc.so.0.9.1.168',
```

```

HOST='my.odbc.host',
PORT=32010,
useEncryption='true',
TOKEN=dremio_access_token,
autocommit=True,
) as conn:
    with conn.cursor() as cursor:
        cursor.execute('select * from test_table')
        results = cursor.fetchall()

```

Users can create OAuth access tokens using a local or LDAP username and password, a PAT, or an external JWT. Dremio provides [sample code](#) for each of these cases.

Downloading and Installing

Windows

Linux

macOS

Downloading and Installing on Windows

note

The Arrow Flight SQL ODBC driver is not available for 32-bit Windows versions.

note

If you plan to use Microsoft Power BI Desktop April 2022 or later to connect to Dremio, you do not need to use this driver. Power BI Desktop April 2022 and later includes a connector that you can use to connect to Dremio. See [Connecting from Microsoft Power BI](#).

To download and install the Arrow Flight SQL ODBC driver:

Download the Windows 64-bit version of the driver from the [ODBC driver download page](#).

Run the installer.

(Optional) In the **User Account Control** page, click **Yes**. This page appears only if there is user account control configured on your Windows machine.

In the **Welcome to Dremio** page, click **Next**.

Click **Install**.

In the **Installation Complete** page, click **Next**.

In the **Completing Arrow Flight SQL ODBC Driver Setup Wizard** page, click **Finish**.

Next, configure the driver.

Downloading and Installing on Linux

To download and install the Arrow Flight SQL ODBC driver:

Download the Linux version of the driver from the [ODBC driver download page](#).

Run the following command to install the driver and automatically create the data source name (DSN) `Arrow Flight SQL ODBC DSN`:

Install driver and create data source name (DSN)

```
sudo yum localinstall <dremio-odbc-rpm-path>
```

Next, configure the driver.

Downloading and Installing on macOS

To download and install the Arrow Flight SQL ODBC driver:

Intel Macs Only

This driver only supports Intel-based Macs. It is not compatible with Apple Silicon M1, M2, and M3 processors.

Download the macOS version of the driver from the [ODBC driver download page](#).

Go to the download location and double-click the downloaded `.dmg` file.

Double-click the `.pkg` file.

In the **Welcome to the ODBC Driver for Arrow Flight SQL Installer** page, click **Continue**.

In the **Standard Install on "Macintosh HD"** page, Click **Install**. Optionally, if you want to change the install location, click **Change Install Location** and navigate to the new location.

In the **Installer is trying to install new software** dialog, specify your macOS password. Then, click **Install Software**.

After the installation is complete, click **Close**.

Next, configure the driver.

Configuring

Configuring on Windows

To configure the System DSN:

danger

Do not follow these steps if you are using Microsoft Power BI Desktop to connect to

Dremio. For the steps for configuring Power BI, see [Connecting from Microsoft Power BI](#).
note

If you want to use a personal access token (PAT), rather than a password, for authenticating to Dremio, generate a PAT. See [Personal Access Tokens](#) for the steps.

Go to **Start Menu > Window Administrative Tools**. Click **ODBC Data Sources (64-bit)**.

In the **ODBC Data Source Administrator (64-bit)** dialog, click **System DSN**.

Select **Arrow Flight SQL ODBC DSN** and click **Configure**.

In the **HOST** field, specify the hostname of the server or its IP address.

In the **PORT** field, specify the port to use for connections from Arrow Flight SQL ODBC client applications, which is 32010 by default.

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
---	---	---	---
UID	Username	Username	Do not specify
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

In the **UseEncryption** field, specify one of these values:

`true`, if Dremio is configured for encrypted communication with your Arrow Flight SQL ODBC client applications.

`false`, if Dremio is not configured for encrypted communication with your Arrow Flight SQL ODBC client applications. Dremio is unencrypted by default.

For additional parameters, see Connection Parameters.

If you ever need to enable tracing for troubleshooting problems with the driver, click the **Tracing** tab in the **ODBC Data Source Administrator (64-bit)** dialog, set the log-file path, and then click **Start Tracing Now**.

Configuring on Linux

note

Before configuring, ensure that unixODBC is installed.

If you want to base your configuration on examples, copy the content of the `odbc.ini` and `odbcinst.ini` files in the `/opt/arrow-flight-sql-odbc-driver/conf` directory and paste the content into your system `/etc/odbc.ini` and `/etc/odbcinst.ini` files.

To configure the properties in the odbc.ini file:

In the **HOST** field, specify the hostname of the server or its IP address.

In the **PORT** field, specify the port to use for connections from Arrow Flight SQL ODBC client applications, which is 32010 by default.

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
---	---	---	---
UID	Username	Username	Do not specify
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

In the **UseEncryption** field, specify one of these values:

``true``, if Dremio is configured for encrypted communication with your Arrow Flight SQL ODBC client applications.

``false``, if Dremio is not configured for encrypted communication with your Arrow Flight SQL ODBC client applications. Dremio is unencrypted by default.

For additional parameters, see Connection Parameters.

note

To find out unixODBC has created your ``odbc.ini`` and ``odbcinst.ini`` files, run this command:

```
odbcinst -j
```

If you ever need to enable tracing for troubleshooting problems with the driver, see the help for unixODBC.

Configuring on macOS

note

Before configuring, ensure that [ODBC Manager](#) is installed.

Launch ODBC Manager.

On the System DSN page, select **Arrow Flight SQL ODBC DSN** and click **Configure**.

(Optional) Change the DSN.

In the **Host** field, specify the hostname of the server or its IP address.

In the **Port** field, specify the port to use for connections from Arrow Flight SQL ODBC client applications, which is 32010 by default.

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
---	---	---	---
UID	Username	Username	Do not specify
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

In the **UseEncryption** field, specify one of these values:

-

``true``, if Dremio is configured for encrypted communication with your Arrow Flight SQL ODBC client applications.

``false``, if Dremio is not configured for encrypted communication with your Arrow Flight SQL ODBC client applications. Dremio is unencrypted by default.

For additional parameters, see Connection Parameters.

If you ever need to enable tracing for troubleshooting problems with the driver, see the help for your driver manager.

Connection Parameters

Primary Connection Parameters

Use these parameters to configure basic connection details such as what data source to connect with.

note

The Arrow Flight SQL ODBC driver does not support password-protected ``.pem`` / ``.crt`` files or multiple ``.crt`` certificates in a single ``.pem`` / ``.crt`` file.

Name	Type	Description	Default Value
Host	string	Sets the IP address or hostname for the Dremio server. If you specify an IP address and you set the TLS connection parameter <code>`useEncryption`</code> to <code>`true`</code> , ensure that the <code>`.etc/hosts/`</code> file includes an entry to map the IP address to the host.	None
Port	integer	Sets the TCP port number that Dremio uses to listen to connections from Arrow Flight SQL ODBC clients.	32010
Schema	string	Provides the name of the database schema to use by default when a schema is not specified in a query. However, this does not prevent queries from being issued for other schemas. Such queries must explicitly include the schema.	None

Specify client information in the appropriate fields for your authentication type:

Field	Username and Password	Personal Access Token	OAuth Access Token
UID	Username	Do not specify	
PWD	Password	Personal access token	Do not specify
TOKEN	Do not specify	Do not specify	OAuth access token

TLS Connection Parameters

Use the following parameters to configure TLS encryption and verification methods for regular connections.

Name	Type	Description	Default Value
useEncryption	integer	Configures the client to use a TLS-encrypted connection to communicate with the Dremio server. Accepted values: <code>* `true`</code> , the client communicates with the Dremio server only using TLS encryption. This is the default value. Therefore, communication between the client application and your Dremio server	

must be encrypted if you do not override this default value. See the configuration of Arrow Flight TLS for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information. * `false`, TLS encryption is disabled with the client. If you specify this value, ensure that the encryption of communication between the client application and your Dremio server is not configured. | true |

| disableCertificateVerification | integer | Specifies whether the driver should verify the host certificate against the trust store. Accepted values: * `false`, the driver verifies the certificate against the trust store. * `true`, the driver does not verify the certificate against the trust store. | false |

| useSystemTrustStore | integer | Controls whether to use a CA certificate from the system's trust store, or from a specified `.pem` file. Accepted values: * `true`, the driver verifies the connection using a certificate in the system trust store. * `false`, the driver verifies the connection using the `.pem` file specified by the `trustedCerts` parameter. | `true` on Windows and macOS, `false` on Linux (which does not have a system truststore) |

| trustedCerts | string | The full path of the `.pem` file containing certificates trusted by a CA, for the purpose of verifying the server. If this option is not set, the driver defaults to using the trusted CA certificates `.pem` file installed by the driver. The exact file path varies according to the operating system on which the driver is installed. The path for the Windows driver differs from the path set for the macOS driver. The TLS connection fails if you do not specify a value when `useEncryption` is `true` and `disableCertificateVerification` is `false`. | N/A |

| hideSQLTablesListing | boolean | Prevents Microsoft Excel 16.95+ from crashing by hiding the list of available sources in Microsoft Excel's Query Dialog. Set to `true` to enable. Only for Intel-based Mac computers. | `false` |

Advanced Parameters

Name	Type	Description	Default Value
---	---	---	---
quoting	string	Specifies which type of character to use to delimit values in queries. The value can be BACK_TICK, BRACKET, or DOUBLE_QUOTE. DOUBLE_QUOTE	DOUBLE_QUOTE
routing_queue	string	Specifies the queue to route queries to during a session. Direct Routing is used to specify the exact queue and execution cluster to run queries on for a given ODBC session. With Direct Routing, workload-management (WLM) rules are not considered; instead, queries are routed directly to the specified queue. For more information, see Workload Management . N/A	N/A
routing_tag	string	When this parameter is set, the specified tag is associated with all queries executed within a session. Rules can check for the presence of a tag with the function "tag()". For more information, see Workload Management . N/A	N/A
stringColumnLength	string	The maximum length of data in columns of the STRING datatype and of complex datatypes. The range is 1 to 2147483647. 1024.	1024

Logging

You can inspect and extract client-side driver logs through the macOS logging infrastructure, because every macOS installation comes with an embedded Console application that allows you to filter by log types. This feature is only available for Intel-based Mac computers, and the log activity of the ODBC driver mostly consists of ODBC API calls.

To start logging:

Open the Console application.

In the search box, select **Excel** in the **PROCESS** dropdown, and **odbc** in the **ANY** dropdown.

!Filter by Excel and ODBC for logs in the Console application.

Click **Start**.

Supported Conversions from Dremio Datatypes to ODBC Datatypes

Dremio Data Types	SQL_\C_BINARY	SQL_\C_BIT	SQL_\C_CHAR	SQL_\C_WCHAR	SQL_\C_TINYINT	SQL_\C_UTINYINT	SQL_\C_SSHORT	SQL_\C_USHORT	SQL_\C_SLONG	SQL_\C_ULONG	SQL_\C_SBIGINT	SQL_\C_UBIGINT	SQL_\C_FLOAT	SQL_\C_DOUBLE	SQL_\C_NUMERIC	SQL_\C_DATE	SQL_\C_TIME	SQL_\C_TIMESTAMP	SQL_\C_GUID	SQL_\C_INTERVAL_*	
BOOLEAN	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
Invalid	N																				
VARBINARY	Y	Invalid	N	N	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
DATE	N	Invalid	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
FLOAT	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
Invalid	N																				
DECIMAL	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Invalid	Invalid	Invalid	Invalid
DOUBLE	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
Invalid	N																				
INTERVAL (day to seconds)	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
INTERVAL (years to months)	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
INT	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
Invalid	N																				
BIGINT	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid	Invalid
Invalid	N																				
TIME	N	N	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
TIMESTAMP	N	N	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
VARCHAR	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Invalid	Invalid
STRUCT	N	N	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
LIST	N	Invalid	Y	Y	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

Was this page helpful?

[Previous

Arrow

Flight

SQL

JDBC](/current/client-applications/drivers/arrow-flight-sql-jdbc-driver)[Next

Dremio JDBC Driver (Legacy)](/current/client-applications/drivers/jdbc)

Supported Operating Systems

Authentication Methods

Username and Password

Personal Access Tokens Enterprise

OAuth Access Tokens Enterprise

Downloading and Installing

Downloading and Installing on Windows

Downloading and Installing on Linux

Downloading and Installing on macOS

Configuring

Configuring on Windows

Configuring on Linux

Configuring on macOS

Connection Parameters

Primary Connection Parameters

TLS Connection Parameters

Advanced Parameters

Logging

Supported Conversions from Dremio Datatypes to ODBC Datatypes

Source: dremio-software-data-products.md

Dremio Software - Data Products

Source:
<https://docs.dremio.com/current/data-products/>

Version: current [26.x]

On this page

Build Data Products

A data product is a self-contained data asset that has been prepared, can be trusted,

and has an SLA. It is meant to optimize for reuse, consistency, and sharing of data. Organizations have data products that are domain-specific and data products that are common at the organization level.

In Dremio, there are two levels of data products:

Data products that are created by curating and transforming one or more source tables into a view.

Data products that are created at the business or application layer and are used in reports, dashboards, and other applications.

Here are two examples of data products:

Sam is a data analyst in the sales team, and he is asked to create a data product that provides a unified data asset that segments customers into different groups based on behavior, spending, and demographics. He creates a data product with a curated and transformed view of customer data from multiple source tables (e.g., customer demographics, purchase history, and engagement activity). The data in the data product is cleansed, transformed, enriched, certified, and structured for easy consumption by marketing and sales teams.

Alice is an analyst in the finance team, and she needs to create an executive financial dashboard at the business or application layer for her company's E-staff team. She creates a data product that powers a visualization report for financial performance metrics for her company, including revenue, expenses, and profitability over time. The data product integrates multiple data sources from various domains, including transactional and financial data, and presents the information in a format that is easily consumable by executives for decision-making.

Data Product Lifecycle

Data products are created and managed through a lifecycle similar to the software development lifecycle. The data product lifecycle encompasses the end-to-end process of developing, deploying, and maintaining data products, ensuring they provide ongoing value to users. The image below depicts the phases of the lifecycle, which teams iterate through.

[!This image illustrates the data products lifecycle.](#)

Discover

To make data products reusable, they must be easily discoverable and explorable. Cross-functional or organization level data products must be accessible across various team to drive consistency. Users need to be able to quickly understand how to interpret the data in a data product and determine if it is relevant to the business problem at hand. For this to happen, data products must be published with adequate metadata such as domain, descriptions and definitions, tags, and usage information. To learn more about data discovery, see [Discover Data](#).

Develop

Data products can be developed using SQL in Dremio's SQL Runner. You can just as

easily use your IDE of choice for development. To learn more about developing data products, see [Develop Data Products](#).

Deploy

To learn more about deploying your semantic layer with dbt, see [Deploy with dbt](#).

Govern

Effective data governance ensures secure, compliant, and transparent management of data by ensuring documentation and traceability, enforcing fine-grained access policies, and tracking dataset lineage to enhance data quality, minimize risks, and optimize value. To learn more about governance, see [Govern Data](#).

Serve

Data products can be served in multiple forms. For example, data products are served in the Dremio console for adhoc analysis or they can be incorporated into a dashboard or report. This phase of the lifecycle is focused on delivering insights and data output to users ensuring accessibility and usability. For more information on the client applications that support connectivity to Dremio, see [Connecting Client Applications to Dremio](#).

Observe

To fully empower users to manage their data products, they must be able to monitor them continuously to assess usage and performance over time. Observability enables data product owners to make iterative improvements based on their users' needs.

Additional Resources

Find out more about data products by enrolling in the [Data Product Fundamentals course in Dremio University](#).

Was this page helpful?

[\[Previous](#)

[Clustering\]\(/current/load-data/clustering\)\)](#)[\[Next](#)

[Discover Data\]\(/current/data-products/discover/\)](#)

[Data Product Lifecycle](#)

[Discover](#)

[Develop](#)

[Deploy](#)

[Govern](#)

Serve

Observe

Additional Resources

Source:
<https://docs.dremio.com/current/data-products/discover/>

Version: current [26.x]

On this page

Discover Data

Dremio simplifies the discovery of data objects and other Dremio entities.

Data discovery capabilities in Dremio include the following:

Catalog exploration in the Dremio console

AI-enabled semantic search for objects and entities

Star objects

Data discovery capabilities are optimized for data that is governed by Dremio. Files and folders that have not been formatted as a table in Dremio may not be easily discoverable.

Catalog exploration in the Dremio console

The Datasets page allows you to navigate through and explore objects that you have access to in Dremio. To learn more about the Datasets page, see [Quick Tour of the Datasets Page](#). Once you have located the table or view that you are interested in, you can use the Details panel to learn more about it.

AI-enabled semantic search for objects Enterprise

You can use the search bar in Dremio to find objects that are accessible through Dremio. This capability performs keyword and semantic search to find tables, views, and other objects that are related to your search criteria. Search only returns results that you have privileges to see. See [Searching for Dremio Objects](#).

Object Metadata

Metadata such as the owner of the table or view, when it was created, and columns is easily accessible through metadata cards. Metadata cards can help you learn more about the object of interest and provide quick links to actions that you may want to take on the table or view. See [Metadata Cards](#) for more information.

Star objects

You can star objects that you use frequently to provide easier discoverability and access. Starring objects such as sources, spaces, folders, tables, and views will surface them in the **Starred** tab of the SQL Runner. See [Star Objects](#) to learn more.

Was this page helpful?

[\[Previous](#)

[Build Data Products\]\(/current/data-products/\)\[Next](#)

[Searching for Dremio Objects\]\(/current/data-products/discover/semantic-search\)](#)

[Catalog exploration in the Dremio console](#)

[AI-enabled semantic search for objects Enterprise](#)

[Object Metadata](#)

[Star objects](#)

Source:
<https://docs.dremio.com/current/data-products/develop>

Version: current [26.x]

On this page

Develop Data Products

You can curate and transform your data to create a data product by

[Writing SQL in the SQL Runner](#)

[Using the low-code transformation flows in the SQL Runner to help generate SQL](#)

Write SQL in the SQL Runner

You can use the SQL Runner to transform your data and create data products in Dremio.

For a quick tour of the SQL Runner and the supported capabilities, see [Quick Tour of the SQL Runner](#).

See the [SQL Reference](#) for functions and commands that you can use to transform and work with your data.

You can also create data products using an IDE of your choice. Use [Arrow Flight JDBC](#) to establish a connection to Dremio from an IDE to do your development.


Create a View

You can create a view from an existing table or view by transforming the data as required and performing the following steps:

Compose the query in the SQL Runner and click **Run** to validate the query. After running the query, click the arrow next to **Save Script As** in the top right of the SQL editor, and select **Save View as...** from the drop-down menu.

Name the new view and select where the view will be located. If the location is not updated, the view will get saved to your home space. Once the view is saved, you will be navigated to the Dataset page.

Retrieve a View Definition

If you have the ``SELECT`` privilege for a view, you can see the underlying definition in the SQL editor but cannot edit the view directly. To see a view definition, on the Datasets page, hover over the line containing the view and click  on the right.


tip

If you have the required privileges, you can run ``SHOW CREATE VIEW <view_name>`` in the SQL editor to see the view definition. See [SHOW CREATE VIEW](#).

Edit a View

If you have the privileges required to edit a view, you can see and edit the definition of the view.

To edit a view, perform the following steps:

On the Datasets page, hover over the line containing the view and click  on the right. The view definition will open.

Edit the view definition as needed and click **Run** to validate the query.

Click **Save View** in the top right corner of the SQL editor. This will overwrite the current definition of the view with the new definition.

Delete a View

Perform the following steps to delete a view:

On the Datasets page, go to the folder or space where your view is located.

Hover over the line containing the view that you want to delete, click the ellipses (...) icon. From the list of actions, click **Delete**. Confirm that you want to delete the view.

caution

If you are deleting a table or view with children, you get a warning. Removing a table or view with children leaves you with disconnected views that you can no longer query.

Use Low-Code to Transform Data in the SQL Runner

To begin a data transformation, via one of the following methods:

Highlighting a portion or all of a field's value

Using the dropdown menu for the transformation.
The dropdown is to the right of the field's name.

Use Highlighting

The highlighting method is often the most intuitive method. It provides enough context for Dremio to make some best guesses about how to execute the transformation that you have in mind.

For instance, you could highlight a portion of a field that contains customer names to quickly perform an extract that creates a new field with only last names.

Suggestion Cards

For transformations that are initiated by highlighting part or all of a field value, Dremio uses a heuristic to determine a set of "suggestion cards" that represent its best guesses as to your intended result.

You can click on one of these suggestion cards to inspect a preview of the new dataset and confirm that it matches your expectations.

If no suggestion card is a perfect match, you can "flip" the card (by clicking the pencil icon in the upper right corner) to tweak the card's parameters before applying the transformation.

note

The highlight method is great for beginning an extract. However, in cases where other capabilities are required, the dropdown menu may be more useful.

Use Dropdown Menus

The dropdown menu provides a more complete list of transformations that are applicable to the data type.

Fix Inconsistent Data with a Join

In situations where the entries in a field are inconsistent (for example, different spellings or abbreviations for the same name), the following technique can be used to increase the quality of the dataset:

Identify the field with the problematic data. It may be useful to run this command in the SQL Editor:

Identify a field

```
SELECT DISTINCT myProblemFieldName FROM myDatasource.myTable
```

Download the results as CSV using the Download button.

Open the file in a text editor or Excel and create lookup values for the distinct values from your table in a second column.

For example, standardizing variations in color names to a single canonical name.

Upload this file to your Home space on Dremio

Open this new dataset and hit the Join button located on the left above the field names

Select Custom Join and then the name of the inconsistent dataset you would like to fix, followed by Next

Drag over the name of the left column from your uploaded dataset, and match it with the name of the field you would like to correct in the inconsistent dataset

Apply the Join then drop the old field, renaming the new one to take its place

Save the corrected dataset

Clean Text

For text data, excess whitespace and changing capitalization schemes are two common data cleanliness issues.

Dremio provides two transformations for dealing with these possible inconsistencies:

Trim Whitespace and **Convert Case**.

Handle Invalid, Empty and NULL Values

Empty or NULL text values are best eliminated by using Exclude.

You can initiate this transformation by:

Highlight a value from the field that contains empty or NULL values, and

Select Exclude from the dropdown that appears.

This renders a list of the values in this field, and the frequency at which they occur.

Check the boxes next to the empty and/or NULL values you which to exclude from the dataset and click Apply.

Work with Date Types

You can convert a text type field that contains date information into a proper date type field. This allows you to do more sophisticated analyses in external tools such as spotting by trends by month, year, or quarter.

You can begin this conversion by selecting 'Date & Time' from the type menu located to the left of the text field's name. In the subsequent dialog, select whether the output should be a time, date, date and time. It also gives a few default options for the format as well as a 'Custom' field for indicating a custom format. See [Data & Time Data Types](#) for more information on the conversions you can do.

Was this page helpful?

[Previous

Star Objects](/current/data-products/discover/bookmarks)[Next

Deploy with dbt](/current/data-products/deploy-with-dbt)

Write SQL in the SQL Runner

Create a View

Retrieve a View Definition

Edit a View

Delete a View

Use Low-Code to Transform Data in the SQL Runner

Use Highlighting

Use Dropdown Menus

Fix Inconsistent Data with a Join

Clean Text

Handle Invalid, Empty and NULL Values

Work with Date Types

Source:
<https://docs.dremio.com/current/data-products/deploy-with-dbt>

Version: current [26.x]

On this page

Deploy with dbt

dbt enables analytics engineers to develop and manage semantic layers within dbt projects and deploy them to Dremio.

You can use Dremio's dbt connector `dbt-dremio` to transform data that is in data sources that are connected to a Dremio project.

Prerequisites

Download the `dbt-dremio` package from <<https://github.com/dremio/dbt-dremio>>.

Ensure that Python 3.9.x or later is installed.

Ensure that you are using Dremio Software version 22.0 or later.

If you want to use TLS to secure the connection between dbt and Dremio Software,

configure full wire encryption in your Dremio cluster. For more information, see the configuration of TLS for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#).

Installing

Install this package from PyPi by running this command:

Install dbt-dremio package

```
pip install dbt-dremio
```

note

`dbt-dremio` works with dbt-core versions 1.8 and 1.9. Earlier versions of dbt-core are out of support from dbt.

Initializing a dbt Project

Run the command `dbt init <project_name>`.

Select `dremio` as the database to use.

Select one of these options to generate a profile for your project:

`software_with_username_password` for working with a Dremio Software cluster and authenticating to the cluster with a username and a password

`software_with_pat` for working with a Dremio Software cluster and authenticating to the cluster with a personal access token

Next, configure the profile for your dbt project.

Profiles

When you initialize a dbt project, you create one of these three profiles. You must configure it before trying to connect to Dremio Cloud or Dremio Software.

Profile for Dremio Software with Username/Password Authentication

Profile for Dremio Software with Authentication Through a Personal Access Token

For descriptions of the configurations in these profiles, see Configurations.

Dremio Software Profile with Username & Password

Example Profile

```
[project name]:
  outputs:
    dev:
      password: b9JtkIgI3uup9gGxxK
      port: 9047
      software_host: 192.0.2.0
```

```

object_storage_source: Samples
object_storage_path: "samples.dremio.com"."Dremio University"
dremio_space: Space1
dremio_space_folder: Folder1.Folder2
threads: 1
type: dremio
use_ssl: true
user: userName
target: dev

```

Dremio Software Profile with Personal Access Token

Example Profile

```

[project name]:
  outputs:
    dev:
      pat: A1BCDrE2FwgH3IJKLM4NoPqrsT5uV6WXYza7I8bcDEFgJ9hIj0Kl1MNOPq2Rstu
      port: 9047
      software_host: 192.0.2.0
      object_storage_source: Samples
      object_storage_path: "samples.dremio.com"."Dremio University"
      dremio_space: Space1
      dremio_space_folder: Folder1.Folder2
      threads: 1
      type: dremio
      use_ssl: true
      user: userName
      target: dev

```

Configurations

Configuration	Required?	Default Value	Description
--- --- --- ---			
`password`	Yes, if you are not using the pat configuration.	None	The password of the account to use when logging into the Dremio cluster.
`pat`	Yes, if you are not using the user and password configurations.	None	The personal access token to use for authenticating to Dremio. See Personal Access Tokens for instructions about obtaining a token. The use of a personal access token takes precedence if values for the three configurations user, password and pat are specified.
`port`	Yes	`9047`	Port for Dremio Software cluster API endpoints.
`software_host`	Yes	None	The hostname or IP address of the coordinator node of the Dremio cluster.
`enterprise_catalog_namespace`	No	None	The name of the catalog in which to create tables, materialized views, tests, and other objects, and views. The dbt aliases are `datalake` (for objects) and `database` (for views). This name corresponds to the name of a catalog in the Open Catalogs section of the Datasets page in Dremio.
`enterprise_catalog_folder`	No	None	The path in the catalog in which to create objects / views. The dbt aliases are `root_path` (for objects) and `schema` (for views). Nested folders in the path are separated with periods. This value corresponds to the

path in this location in the Datasets page in Dremio. |
`object_storage_source`	No	\$scratch	The name of the filesystem in which to create tables, materialized views, tests, and other objects. The dbt alias is `datalake`. This name corresponds to the name of a source in the **Object Storage** section of the Datasets page in Dremio:
`object_storage_path`	No	`no_schema`	The path in the filesystem in which to create objects. The default is the root level of the filesystem. The dbt alias is `root_path`. Nested folders in the path are separated with periods. This value corresponds to the path in this location in the Datasets page in Dremio: 'samples.dremio.com'. 'Dremio University'
`dremio_space`	No	`@<username>`	The value of the Dremio space in which to create views. The dbt alias is `database`. This value corresponds to the name in this location in the **Spaces** section of the Datasets page in Dremio: Spaces1
`dremio_space_folder`	No	`no_schema`	The folder in the Dremio space in which to create views. The default is the top level in the space. The dbt alias is `schema`. Nested folders are separated with periods. This value corresponds to the path in this location in the Datasets page in Dremio:
`threads`	Yes	1	The number of threads the dbt project runs on.
`type`	Yes	dremio	Auto-populated when creating a Dremio project. Do not change this value.
`use_ssl`	Yes	`true`	Acceptable values are `true` and `false`. If the value is set to true, ensure that full wire encryption is configured in your Dremio cluster. See [Prerequisites](#).
`verify_ssl`	No	`true`	Acceptable values are `true` and `false`. Set to `false` if using a self-signed certificate or if the root certificate authority (CA) is not included in Python's CA certificates.
`user`	Yes	None	The username of the account to use when logging into the Dremio cluster.

Known Issues

[Model contracts](#) are not supported.

Additional Resources

Learn more about DataOps by enrolling in the [DataOps with Apache Iceberg course in Dremio University](#).

Was this page helpful?

[\[Previous](#)

[Develop Data Products\]\(/current/data-products/develop\)](#)[\[Next](#)

[Govern Data\]\(/current/data-products/govern/\)](#)

[Prerequisites](#)

[Installing](#)

[Initializing a dbt Project](#)

[Profiles](#)

-

Dremio Software Profile with Username & Password

Dremio Software Profile with Personal Access Token

Configurations

Known Issues

Additional Resources

Source:
<https://docs.dremio.com/current/data-products/govern/>

Version: current [26.x]

On this page

Govern Data

Effective data governance is essential for managing secure, compliant, transparent, and traceable data products. In addition to securing your data using role-based access control (RBAC), data governance for data products allows you to apply fine-grained access controls (row access & column-masking policies) on your data, as well as trace the lineage of any dataset in the lakehouse. Together, these governance tools help organizations optimize the value they derive from their data to enhance data quality and transparency, while minimizing the risks associated with data misuse and non-compliance.

Dremio also provides tools for describing, identifying, and displaying datasets using wikis and tags.

Row-Access and Column-Masking Policies

Row access & column-masking (or fine-grained access control) policies provide mechanisms to enforce data privacy and security rules directly on your data.

Lineage

Data lineage allows you to track and visualize data as it moves through the various stages of a data pipeline. It provides clarity on where the data comes from, how it is transformed, and where it is used.

Wikis and Tags

Wikis for datasets allow users to document and describe datasets in the Open Catalog using a rich text editor with Github-flavored markdown.

Tags for datasets help organize and retrieve data efficiently by allowing users to search, filter, and locate datasets through specific tag associations.

Was this page helpful?

[\[Previous](#)

[Deploy with dbt\]\(/current/data-products/deploy-with-dbt\)\[Next](#)

[Row-Access Policies\]\(/current/data-products/govern/row-column-policies-udf\)](#) and [Column-Masking](#)

[Row-Access and Column-Masking Policies](#)

[Lineage](#)

[Wikis and Tags](#)

Source:
<https://docs.dremio.com/current/data-products/discover>

Version: current [26.x]

On this page

Discover Data

Dremio simplifies the discovery of data objects and other Dremio entities.

Data discovery capabilities in Dremio include the following:

- Catalog exploration in the Dremio console

- AI-enabled semantic search for objects and entities

- Star objects

Data discovery capabilities are optimized for data that is governed by Dremio. Files and folders that have not been formatted as a table in Dremio may not be easily discoverable.

Catalog exploration in the Dremio console

The Datasets page allows you to navigate through and explore objects that you have access to in Dremio. To learn more about the Datasets page, see [Quick Tour of the Datasets Page](#). Once you have located the table or view that you are interested in, you can use the Details panel to learn more about it.

AI-enabled semantic search for objects Enterprise

You can use the search bar in Dremio to find objects that are accessible through Dremio. This capability performs keyword and semantic search to find tables, views, and other objects that are related to your search criteria. Search only returns results that you have privileges to see. See [Searching for Dremio Objects](#).

Object Metadata

Metadata such as the owner of the table or view, when it was created, and columns is easily accessible through metadata cards. Metadata cards can help you learn more about the object of interest and provide quick links to actions that you may want to take on the table or view. See [Metadata Cards](#) for more information.

Star objects

You can star objects that you use frequently to provide easier discoverability and access. Starring objects such as sources, spaces, folders, tables, and views will surface them in the **Starred** tab of the SQL Runner. See [Star Objects](#) to learn more.

Was this page helpful?

[\[Previous](#)

[Build Data Products\]\(/current/data-products/\)\[Next](#)

[Searching for Dremio Objects\]\(/current/data-products/discover/semantic-search\)](#)

[Catalog exploration in the Dremio console](#)

[AI-enabled semantic search for objects Enterprise](#)

[Object Metadata](#)

[Star objects](#)

Source:
<https://docs.dremio.com/current/data-products/govern>

Version: current [26.x]

On this page

Govern Data

Effective data governance is essential for managing secure, compliant, transparent, and traceable data products. In addition to securing your data using role-based access control (RBAC), data governance for data products allows you to apply fine-grained access controls (row access & column-masking policies) on your data, as well as trace the lineage of any dataset in the lakehouse. Together, these governance tools help organizations optimize the value they derive from their data to enhance data quality and transparency, while minimizing the risks associated with data misuse and non-compliance.

Dremio also provides tools for describing, identifying, and displaying datasets using wikis and tags.

Row-Access and Column-Masking Policies

Row access & column-masking (or fine-grained access control) policies provide mechanisms to enforce data privacy and security rules directly on your data.

Lineage

Data lineage allows you to track and visualize data as it moves through the various stages of a data pipeline. It provides clarity on where the data comes from, how it is transformed, and where it is used.

Wikis and Tags

Wikis for datasets allow users to document and describe datasets in the Open Catalog using a rich text editor with Github-flavored markdown.

Tags for datasets help organize and retrieve data efficiently by allowing users to search, filter, and locate datasets through specific tag associations.

Was this page helpful?

[Previous

Deploy with dbt](/current/data-products/deploy-with-dbt)[Next

Row-Access Policies] and Column-Masking Policies](/current/data-products/govern/row-column-policies-udf)

Row-Access and Column-Masking Policies

Lineage

Wikis and Tags

Source: dremio-software-data-sources.md

Dremio Software - Data Sources

Source:
<https://docs.dremio.com/current/data-sources/>

Version: current [26.x]

On this page

Manage Sources

Dremio supports a variety of data sources, including lakehouse catalogs, object storage, and databases.

As the Dremio Shared Responsibility Models outline, metadata is a shared responsibility between Dremio and you. The Shared Responsibility Models lay out Dremio's responsibilities for enabling data source configurations and your responsibilities for managing metadata.

Open Catalog

Dremio comes with a built-in lakehouse catalog, built on [Apache Polaris \(incubating\)](#). The catalog enables centralized, secure read and write access to your Iceberg tables across various REST-compatible query engines and automates data maintenance operations to maximize query performance.

[Open Catalog](#)

Lakehouse Catalogs

Lakehouse catalogs provide you with the ability to connect to centralized catalogs. The Open Catalog, Snowflake Open Catalog, Unity Catalog, and Iceberg REST Catalog all connect to the destination sources over the Apache Iceberg REST API.

[AWS Glue Data Catalog](#)

[Open Catalog \(External\)](#)

[Hive](#)

[Iceberg REST Catalog](#)

[Nessie](#)

[Snowflake Open Catalog](#)

[Unity Catalog](#)

[Microsoft OneLake](#)

Object Storage

[Amazon S3](#)

[Azure Storage](#)

[Google Cloud Storage](#)

[HDFS](#)

[NAS](#)

Databases

[Amazon OpenSearch Service](#)

[Amazon Redshift](#)

[Apache Druid](#)

[Dremio Cluster](#) (you can connect to one or more other Dremio Software clusters and run queries on the data sources to which they are connected, and you can run queries that federate data across connected clusters)

[Google BigQuery](#)

-

[Elasticsearch](#)

[IBM Db2](#)

[Microsoft Azure Data Explorer](#)

[Microsoft Azure Synapse Analytics](#)

[Microsoft SQL Server](#)

[MongoDB](#)

[MySQL](#)

[Oracle](#)

[PostgreSQL](#)

[SAP HANA](#)

[Snowflake](#)

[Teradata](#)

[Vertica](#)

Dremio enables users to run external queries, queries that use the native syntax of the relational database, to process SQL statements that are not yet supported by Dremio or are too complex to convert. Dremio administrators enable the feature for each data source and specify which Dremio users can edit that source. See [Querying Relational-Database Sources Directly](#) for more information.

Dremio improves query performance for relational database datasets with [Runtime Filtering](#), which applies dimension table filters to joined fact tables at runtime.

note

Decimal Support: Decimal-to-decimal mappings are supported for relational database sources.

Collation: Relational database sources must have a collation equivalent to `LATIN1_GENERAL_BIN2` to ensure consistent results when operations are pushed down. For non-equivalent collations, create a view that coerces the collation to one that is equivalent to `LATIN1_GENERAL_BIN2` and access that view.

For all sources, case-sensitive source data file/table names are not supported. In Dremio, case is ignored in the names of data files. `file1.parquet`, `File1.parquet`, and `FILE1.parquet` are considered to be equivalent names. Therefore, searching on one of these names can result in unanticipated results.

In addition, columns in a table that have the same name with different cases are not supported. For example, if two columns named `Trip_Pickup_DateTime` and `trip_pickup_datetime` exist in the same table, one of the columns may disappear when the header is extracted.

Files and Folders

[Formatting Data to a Table](#)

Upload Files

note Case-sensitive source data file/table names are not supported. In Dremio, data filenames in your data source are "seen" in a case-insensitive manner. So, if you have three file names with difference cases (for example, `JOE`, `Joe`, and `joe`), Dremio "sees" the files as having the same name. Thus, searching on `Joe`, `JOE`, or `joe`, can result in unanticipated data results.

In addition, columns in a table that have the same name with different cases are not supported. For example, if two columns named `Trip_Pickup_DateTime` and `trip_pickup_datetime` exist in the same table, one of the columns may disappear when the header is extracted.

Was this page helpful?

[Previous

AI](/current/ai)[Next

Open Catalog](/current/data-sources/open-catalog/)

Open Catalog

Lakehouse Catalogs

Object Storage

Databases

Files and Folders

Source:
<https://docs.dremio.com/current/data-sources/open-catalog/>

Version: current [26.x]

On this page

Open Catalog Enterprise

Dremio's built-in lakehouse catalog is built on Apache Polaris (incubating). The catalog enables centralized, secure read and write access to your Iceberg tables across different REST-compatible query engines and automates data maintenance operations to maximize query performance. Key features include:

Iceberg REST compatibility: Read and write from the Open Catalog using any engine or framework compatible with the Iceberg REST API. For example, use Spark or Flink to ingest data into the catalog, and then use Dremio to curate and serve data products built on that data.

Role-Based Access Control and Fine-Grained Access Control: Secure data using Role-Based Access Control (RBAC) privileges and create row filters and column masks to ensure users only access the data they need. For example, create a column mask to obfuscate credit card numbers or create a row filter on your employee details table that

only returns rows with employees in your region.

Automated table maintenance: Open Catalog automates Iceberg maintenance operations like compaction and vacuum, which maximizes query performance, minimizes storage costs, and eliminates the need to run manual data maintenance. Open Catalog also simplifies Iceberg table management and eliminates the risk of poor performance from suboptimal data layouts with support for Iceberg clustering keys.

Enable data analysts: Open Catalog is fully compatible with Dremio's built-in data product capabilities, including semantic search (use natural language to discover AI-ready data products), descriptions (use built-in descriptions and labels to understand how to use data products to answer business questions), and lineage (use lineage graphs to understand how data products are derived and transformed and assess the impact of changes on downstream datasets).

This page provides instructions for configuring the Open Catalog. If you would like to connect to Open Catalogs deployed in other Dremio instances, see [Open Catalog \(External\)](#).

Prerequisites

Before you configure Open Catalog, make sure you do the following:

Configure access to your storage provider, as described in [Configure Storage Access](#).

Configure the settings of your storage provider in Dremio's Helm chart, as described in [Configuring Storage for the Open Catalog](#).

These configurations are required to enable support for vended credentials and to allow access to the table metadata necessary for Iceberg table operations.

Configure the Open Catalog

To configure Open Catalog:

When creating the first Open Catalog, select **Add an Open Catalog**. Add a **Name** for the catalog.

When configuring an existing Open Catalog, right-click on your catalog and select **Settings** from the dropdown.

Storage

The **Default storage URI** field displays the default storage location you configured in [Dremio's Helm chart](#).

Use the **Storage access** field to configure your preferred authentication method. Open Catalog supports two types of credentials for authentication:

Use credential vending (Recommended): Credential vending is a security mechanism where the catalog service issues temporary, scoped access credentials to the query engine for accessing table storage. The engine is "vended" a temporary credential just in time for the query.

Use master storage credentials: The credentials authenticate access to all storage URIs within this catalog. These credentials ensure all resources are accessible through a single authentication method. This should be used if STS is unavailable or the vended credentials mechanism is disabled. Select the object storage provider that hosts the location specified in the **Default storage URI** field:

AWS – Select **AWS** for Amazon S3 and S3-compatible storage. You can refer to the Dremio documentation for connecting to [Amazon S3](#), which is also applicable here. When selecting to assume an IAM role, ensure that the role policy grants access to the bucket or folder specified in the **Default storage URI** field.

Azure – Select **Azure** for Azure Blob Storage. You can refer to the Dremio documentation for connecting to [Azure Storage](#), which is also applicable here.

Google Cloud Storage – Select **Google** for Google Cloud Storage (GCS). You can refer to the Dremio documentation for connecting to [GCS](#), which is also applicable here.

Enter any required storage connection properties in the **Connection Properties** field. Refer to the Advanced Options section for your storage provider (Amazon S3, Azure, or GCS) for available properties.

Advanced Options

To set advanced options:

Under **Cache Options**, review the following table and edit the options to meet your needs.

Cache Options	Description
--- ---	
Enable local caching when possible	Selected by default. Along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.
Max percent of total available cache space to use when possible	Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter a percentage in the value field or use the arrows to the far right to adjust the percentage.

Under **Table maintenance**, manage settings for [automated table maintenance operations](#):

Enable auto optimization: Compacts small files into larger files. Clusters data if Iceberg clustering keys are set on the table.

Enable table cleanup: Deletes expired snapshots and orphaned metadata files.

Reflection Refresh

You can set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed. See the following options:

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh. The default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days, or weeks. This option is ignored if Never refresh is selected.
Set refresh schedule	Specify the daily or weekly schedule.
Never expire	Select to prevent Reflections from expiring. The default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days, or weeks. This option is ignored if Never expire is selected.

Metadata

Specifying metadata options is handled with the following settings:

Dataset Handling

Remove dataset definitions if the underlying data is unavailable (default).

If this box is *not* checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names, such as databases and tables. Set the time interval using this parameter.

Parameter	Description
---	---
Fetch every	You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning, such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

Parameter	Description
---	---
Fetch mode	You can choose to fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to Only Queried Datasets .
Fetch every	You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.
Expire after	You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.

Privileges

You have the option to grant privileges to specific users or roles. See [Access Control](#) for additional information about privileges.

To grant access to a user or role:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Configure Storage Access

To configure access to the storage, select your storage provider below and follow the steps:

Amazon S3

S3-compatible

Azure Storage

Google Cloud Storage

S3 and STS Access via IAM Role (Preferred)

Create an Identity and Access Management (IAM) user or use an existing IAM user for Open Catalog.

Create an IAM policy that grants access to your S3 location. For example:

Example of a policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion"
      ],
      "Resource": "arn:aws:s3:::<my_bucket>/*"
    },
    {
      "Effect": "Allow",
      "Action": [
```

```

    "s3:ListBucket",
    "s3:GetBucketLocation"
  ],
  "Resource": "arn:aws:s3:::<my_bucket>",
  "Condition": {
    "StringLike": {
      "s3:prefix": [
        "*"
      ]
    }
  }
}
]
}

```

Create an IAM role to grant privileges to the S3 location.

In your AWS console, select **Create Role**.

Enter an **externalId**. For example, ``my_catalog_external_id``.

Attach the policy created in the previous step and create the role.

Create IAM user permissions to access the bucket via STS:

The `sts:AssumeRole` permission is required for Open Catalog to function with vended credentials, as it relies on the STS temporary token to perform these validations.

Select the IAM role created in the previous step.

Edit the trust policy and add the following:

Trust policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "AWS": "<dremio_catalog_user_arn>"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "<dremio_catalog_external_id>"
        }
      }
    }
  ]
}
```


Replace the following values with the ones obtained in the previous steps:

`<dremio_catalog_user_arn>` - The IAM user that was created in the first step.

`<dremio_catalog_external_id>` - The external ID that was created in the third step.

S3 and STS Access via Access Key

In the Dremio console, select **Use master storage credentials** when adding Open Catalog.

The access keys must have permissions to access the bucket and the STS server.

Create a Kubernetes secret named `catalog-server-s3-storage-creds` to access the configured location. Here is an example for S3 using an access key and secret key:

Run kubectl to create the Kubernetes secret

```
export AWS_ACCESS_KEY_ID=<username>
export AWS_SECRET_ACCESS_KEY=<password>
kubectl create secret generic catalog-server-s3-storage-creds \
--namespace $NAMESPACE \
--from-literal awsAccessKeyId=$AWS_ACCESS_KEY_ID \
--from-literal awsSecretAccessKey=$AWS_SECRET_ACCESS_KEY
```

The access keys must have permissions to access the bucket.

To use vended credentials, the access key must also have access to an STS server.

If you cannot leverage an STS server, when setting up the catalog for the first time in the Dremio console, you must select master storage credentials.

Create a Kubernetes secret named `catalog-server-s3-storage-creds` to access the configured location. Here is an example for S3 using an access key and secret key:

Run kubectl to create the Kubernetes secret

```
export AWS_ACCESS_KEY_ID=<username>
export AWS_SECRET_ACCESS_KEY=<password>
kubectl create secret generic catalog-server-s3-storage-creds \
--namespace $NAMESPACE \
--from-literal awsAccessKeyId=$AWS_ACCESS_KEY_ID \
--from-literal awsSecretAccessKey=$AWS_SECRET_ACCESS_KEY
```

For S3-compatible storage providers (e.g., MinIO), the access keys should be the username and password.

note

Soft delete for blobs is not supported for Azure Storage accounts. Soft delete should be disabled to establish a successful connection.

Although not mandatory, Dremio recommends enabling **Hierarchical Namespace** when using Azure Data Lake Storage. For more information, see [Azure Data Lake](#)

Storage Gen2 hierarchical namespace.

Register an application and create secrets:

Go to Azure Active Directory > App Registrations.

Register your app and take note of the **Client ID** and **Tenant ID**. For more information on these steps, refer to [Register an application with Microsoft Entra ID and create a service principal](#).

Go to Certificates & Secrets > New Client Secret.

Create a secret and take note of the **Secret Value**.

Create a Kubernetes secret named `catalog-server-azure-storage-creds` using the following command:

Run kubectl to create the Kubernetes secret

```
export AZURE_CLIENT_ID=<Azure App client id>
export AZURE_CLIENT_SECRET=<App secret value>
kubectl create secret generic catalog-server-azure-storage-creds \
  --namespace $NAMESPACE \
  --from-literal azureClientId=$AZURE_CLIENT_ID \
  --from-literal azureClientSecret=$AZURE_CLIENT_SECRET
```

Create an Identity and Access Management (IAM) role in your Storage Account and set up the permission for your new application to access the storage account by following these steps:

In the Azure console, go to your Storage Account and navigate to Access Control (IAM) > Role assignments > Add role assignment.

Select the `Storage Blob Data Contributor` role and click **Next**.

In the **Members** section, click **Select members**, search for your app registration from step 1, and click **Select**.

Review and assign the roles.

Go to your Google Cloud Platform (GCP), [create a service account](#), and grant an Identity and Access Management (IAM) role with the following permissions:

Permissions for the IAM role

```
storage.buckets.get
storage.objects.create
storage.objects.delete
storage.objects.get
storage.objects.list
```

Obtain the JSON file with the GCP credentials from the Google service account.

Create the Kubernetes secret where Dremio is deployed using the following command:

Run kubectl to create the Kubernetes secret

```
kubectl create secret generic catalog-server-gcs-storage-creds --from-file=<filename>.json
```

Update an Open Catalog Source

To update an Open Catalog source:

On the Datasets page, in the panel on the left, find the name of the Open Catalog source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the Source Settings dialog, edit the settings you wish to update. Dremio does not support updating the source name.

Click **Save**. Once you have configured Open Catalog, the Catalog REST APIs are accessible via `http://{DREMIO_ADDRESS}:8181/api/catalog`, where ``DREMIO_ADDRESS`` is the IP address of your Dremio cluster.

Using the Open Catalog with Multiple Storage Locations

You can use one Open Catalog instance to work with data stored in multiple storage buckets. For example, you can create different folders (namespaces) in one Open Catalog instance, such that data in Folder A is stored in Storage Bucket 1, and data in Folder B is stored in Storage Bucket 2. This feature is named **Storage URIs (Uniform Resource Identifiers)**.

A Storage URI is an optional attribute that can be attached to a folder and consists of a path to an object storage location. When you create a folder, you can either configure the folder to use the "inherited" storage location you defined when you configured Open Catalog or when you set the Storage URI on one of its parent folders, or you can configure the folder to use a custom Storage URI. To configure the folder to use a custom Storage URI, add the path to the object storage location you would like to use during folder creation. Ensure that the storage credentials you are using for the Open Catalog can access the object storage location you added for your newly created folder.

Storage URIs Example

The diagram below depicts an Open Catalog that contains two namespaces (``NS1``, ``NS2``), where its underlying folders utilize Storage URIs to store data in custom storage locations:



In this example:

TBL1 would be stored in `<Uri1>/NS3/TBL1``

TBL3 would be stored in ``<Uri2>/NS5/NS6/TBL3``

TBL4 would be stored in ``<Default URI>/NS2/TBL4``

When creating a table from an external Open Catalog source, the default Storage URI that the table will use is the root path of the external Open Catalog source, unless one of the folders on the table's path has been set with a custom Storage URI.

Was this page helpful?

[Previous

Manage Sources](/current/data-sources/)[Next

Connect to Open Catalog from Apache Spark](/current/data-sources/open-catalog/connecting-from-spark)

Prerequisites

Configure the Open Catalog

Storage

Advanced Options

Reflection Refresh

Metadata

Privileges

Configure Storage Access

S3 and STS Access via IAM Role (Preferred)

S3 and STS Access via Access Key

Update an Open Catalog Source

Using the Open Catalog with Multiple Storage Locations

Storage URIs Example

Source:
<https://docs.dremio.com/current/data-sources/lakehouse-catalogs/>

Version: current [26.x]

Lakehouse Catalogs

The following lakehouse catalogs are supported as data sources in Dremio:

Open Catalog (External)

AWS Glue Data Catalog

[Microsoft OneLake](#)

[Snowflake Open Catalog](#)

[Unity Catalog](#)

[Iceberg REST Catalog](#)

[Hive](#)

[Nessie](#)

Was this page helpful?

[\[Previous](#)

[Connect to Open Catalog from Apache Spark\]\(/current/data-sources/open-catalog/connecting-from-spark\)\[Next](#)

[Open Catalog \(External\)\]\(/current/data-sources/lakehouse-catalogs/open-catalog-external\)](#)

Source:
<https://docs.dremio.com/current/data-sources/object/>

Version: current [26.x]

Object Storage

You can run queries directly on the data in your data lake by formatting directories and files into tables. The following types of object storage are supported:

[Amazon S3](#)

[Azure Storage](#)

[Google Cloud Storage](#)

[HDFS](#)

[NAS](#)

Was this page helpful?

[\[Previous](#)

[Lakehouse Catalogs\]\(/current/data-sources/lakehouse-catalogs/\)\[Next](#)

[Amazon S3\]\(/current/data-sources/object/s3\)](#)

Source:
<https://docs.dremio.com/current/data-sources/databases/>

Databases

You can run queries directly against relational databases, NoSQL databases, and data warehouses, referred to in Dremio as databases. Using these databases, you can perform “external queries.” External queries use native syntax for a source and can be used for SQL statements that are not supported directly in Dremio or are too complex to convert.

The following databases are supported:

[Amazon OpenSearch Service](#)

[Amazon Redshift](#)

[Apache Druid](#)

[Dremio Cluster](#)

[Google BigQuery](#)

[Elasticsearch](#)

[IBM Db2](#)

[Microsoft Azure Data Explorer](#)

[Microsoft Azure Synapse Analytics](#)

[Microsoft SQL Server](#)

[MongoDB](#)

[MySQL](#)

[Oracle](#)

[PostgreSQL](#)

[SAP HANA](#)

[Snowflake](#)

[Teradata](#)

[Vertica](#)

Was this page helpful?

[\[Previous](#)

[Object Storage\]\(/current/data-sources/object/\)](#)[\[Next](#)

[Amazon OpenSearch Service\]\(/current/data-sources/databases/opensearch\)](#)

Source:

Version: current [26.x]

On this page

Upload Files

You can upload files to your home space for experimentation. Excel, JSON, Parquet, and CSV files are supported for upload. Once it is uploaded, you can query the file just like a table by specifying the path, "@[home-space-name].[file-name]".

Upload File to Your Home Space

In the Dremio console, navigate to your home space by clicking on the home icon and your username.

Click [!This is the Add icon to upload a file or add a folder.](#) in the upper right side and select **Upload File**.

Select the file and click **Next**. If the file is large, it may take a few moments to upload, depending on your connection speed.

(Optional) During the upload process, configure the file settings. For example, configure how the file is delimited.

Click **Save**.

Once the file has been uploaded, it is displayed in your home space as a table. You can query it by running ``SELECT * FROM "@username"."table_name"``.

Limits

Uploaded files are copies of your local file. Updates to your local file are not automatically reflected in Dremio.

Bulk upload of multiple files is not supported.

Files uploaded to your home space cannot be shared with other users. To share it with others, upload the file into a shared source or use [COPY INTO](#) to create an Iceberg table in your Open Catalog.

Was this page helpful?

[\[Previous](#)

[Vertica\]\(/current/data-sources/databases/vertica\)](#)[\[Next](#)

[Load Data\]\(/current/load-data/\)](#)

[Upload File to Your Home Space](#)

[Limits](#)

Source:
<https://docs.dremio.com/current/data-sources/warehouse-catalogs/aws-glue-catalog/>

Version: current [26.x]

On this page

AWS Glue Data Catalog

Dremio supports Amazon S3 datasets cataloged in [AWS Glue](#) as a Dremio data source. Files in S3 must be one of the following file formats or table formats:

Apache Iceberg

Delimited text files (CSV/TSV)

Delta Lake (Dremio supports reading Native Delta Lake tables in AWS Glue. Delta Lake symlink tables must be crawled and native Delta Lake tables created from them. See [Introducing native Delta Lake table support with AWS Glue crawlers](#) in the AWS Big Data blog.)

ORC

Parquet

AWS Glue data sources added to projects default to using the Apache Iceberg table format. When upgrading, AWS Glue data sources added to projects before Dremio 22 are modified to use the Apache Iceberg table format as the default format.

AWS Glue Credentials

Dremio administrators need credentials to access files in Amazon S3 and list databases and tables in the AWS Glue Catalog. Dremio recommends using the provided sample [AWS managed policy](#) when configuring a new AWS Glue Catalog data source. See [Dremio Configuration](#) for more information about supported authentication mechanisms.

Dremio reads the table metadata from AWS Glue and directly scans the data on S3 using its high-performance, massively parallel processing (MPP) engine. For this reason, you need to give permissions to connect to Glue as well as the permissions to read the data on S3 for those tables.

AWS IAM Policy for Accessing Amazon S3 and AWS Glue

Dremio recommends using the following AWS managed policy:

IAM policy for accessing Amazon S3 and AWS Glue

```
{
  "Version": "2012-10-17",
  "Statement": [
    # Allow Dremio to run the listed AWS Glue API operations.
```



```

{
  "Effect": "Allow",
  "Action": [
    "glue:GetDatabase",
    "glue:GetDatabases",
    "glue:GetPartition",
    "glue:GetPartitions",
    "glue:GetTable",
    "glue:GetTableVersions",
    "glue:GetTables",
    "glue:GetConnection",
    "glue:GetConnections",
    "glue:GetDevEndpoint",
    "glue:GetDevEndpoints",
    "glue:GetUserDefinedFunction",
    "glue:GetUserDefinedFunctions",
    "glue:BatchGetPartition"
  ],
  "Resource": [
    "*"
  ]
},
# Allow Dremio to read and write files in a bucket.
{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject",
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::aws-glue-*/*",
    "arn:aws:s3:::*/*aws-glue-*/*"
  ]
},
# Allow Dremio to access the Amazon S3 buckets or folders with names containing
either the 'aws-glue-' or 'crawler-public' prefixes.
{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject"
  ],
  "Resource": [
    "arn:aws:s3:::crawler-public*",
    "arn:aws:s3:::aws-glue-*"
  ]
},
# Allow Dremio to create or delete tags in the AWS Glue catalog.
{
  "Effect": "Allow",
  "Action": [
    "ec2:CreateTags",
    "ec2:DeleteTags"
  ],
  "Condition": {

```

```

        "ForAllValues:StringEquals": {
            "aws:TagKeys": [
                "aws-glue-service-resource"
            ]
        },
        "Resource": [
            "arn:aws:ec2:*:*:network-interface/*",
            "arn:aws:ec2:*:*:security-group/*",
            "arn:aws:ec2:*:*:instance/*"
        ]
    }
}

```

Configuring AWS Glue Data Catalog as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Lakehouse Catalogs**, select **AWS Glue Data Catalog**.

General

Users with proper [privileges](#) can configure access to AWS Glue Catalog with one of the three authentication methods.

Name

Specify a name for the data source. You cannot change the name after the source is created. The name cannot include the following special characters: ``/``, ``:``, ``[``, or ``]``.

AWS Region Selection

Specify a region from which you want to see the tables from AWS Glue. Only tables from this region will be shown after the connection is made.

Authentication

Choose one of the following authentication methods:

AWS Access Key: All or allowed (if specified) buckets associated with this access key or IAM role to assume, if provided, will be available.

Under **AWS Access Key**, enter the [AWS access key ID](#).

Under **AWS Access Secret**, provide the [AWS access secret](#) using one of the following methods:

Dremio: Provide the access secret in plain text. Dremio stores the access secret.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the access secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the access secret reference in the correct format.

Under **IAM Role to Assume**, enter the IAM role that Dremio should assume in conjunction with the AWS Access Key authentication method.

EC2 Metadata: All or allowed (if specified) buckets associated with the specified IAM role attached to EC2 or IAM role to assume, if provided, will be available.

Under **IAM Role to Assume**, enter the IAM role that Dremio should assume in conjunction with the EC2 Metadata authentication method.

EKS Pod Identity: Dremio can access all S3 buckets linked to the IAM role associated with the Kubernetes service account or the assumed IAM role. If you specify certain buckets, only those will be available.

Under **IAM Role to Assume**, enter the IAM role that Dremio should assume when using the Pod Identity authentication method.

AWS Profile: Dremio sources profile credentials from the specified AWS profile. For information on how to set up a configuration or credentials file for AWS, see [AWS Custom Authentication](#).

AWS profile (optional): The AWS profile name. If this is left blank, then the default profile will be used. For more information about using profiles in a credentials or configuration file, see AWS's documentation on [Configuration and credential file settings](#).

The **Encrypt connection** option is enabled by default to encrypt the connection to AWS Glue. Clear the checkbox to disable encryption.

Allowed Databases

The allowed databases configuration is a post-connection filter on the databases visible from AWS Glue. When selective access to the databases within AWS Glue is required, the allowed databases filter will limit access within Dremio to only the needed databases per source connection, thus improving data security and source metadata refresh performance.

When the allowed database filter is empty, all databases from the AWS Glue source are visible in Dremio. When a database is added or removed from the filter, Dremio performs an asynchronous update to expose new databases and remove databases not included in the filter. Each entry in the allowed database filter must be a valid database name; misspelled or nonexistent databases are ignored.

Advanced Options

All configurations are optional.

Connection Properties

A list of additional connection properties that can be specified to use with the connection.

Locations in which Iceberg Tables are Created

Where the CREATE TABLE command creates an Iceberg table depends on the type of data source being used. For AWS Glue Data Sources, the root directory is assumed by default to be `/user/hive/warehouse``. If you want to create tables in a different location, you must specify the S3 address of an Amazon S3 bucket in which to create them:

On the Advanced Options page of the Edit Source dialog, add this connection property: ``hive.metastore.warehouse.dir``.

Set the value to the S3 address of an S3 bucket.

The schema path and table name are appended to the root location to determine the default physical location for a new Iceberg table.

Lake Formation Integration

Lake Formation provides access controls and allows administrators to define security policies. Enabling this functionality and additional details on the configuration options below are described in more detail on the [Integrating with Lake Formation](#) page.

Enforce AWS Lake Formation access permissions on datasets. Dremio checks any datasets included in the AWS Glue source for the required permissions to perform queries.

Prefix to map Dremio users to AWS ARNs. Leave blank to default to the end user's username, or enter a REGEX expression.

Prefix to map Dremio groups to AWS ARNs. Leave blank to default to the end user's group, or enter a REGEX expression.

Reflection Refresh

Specify how frequently Dremio refreshes Data Reflections based on the AWS Glue data source in the ``Reflection Refresh`` tab. Dremio refreshes every hour and expires after three hours by default.

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Specify how and how frequently Dremio refreshes metadata on the ``Metadata`` tab. By default, Dremio fetches top-level objects and dataset details every hour. Dremio

retrieves details only for queried datasets by default to improve query performance.

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an AWS Glue Data Catalog Source

To update an AWS Glue Data Catalog source:

On the Datasets page, under **Lakehouse Catalogs** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring AWS Glue Data Catalog as a Source](#).

Click **Save**.

Deleting an AWS Glue Data Catalog Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an AWS Glue Data Catalog source, perform these steps:

On the Datasets page, click **Sources > Lakehouse Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[Previous

Open Catalog
(External)](/current/data-sources/lakehouse-catalogs/open-catalog-external)[Next

Microsoft OneLake](/current/data-sources/lakehouse-catalogs/onelake)

AWS Glue Credentials

AWS IAM Policy for Accessing Amazon S3 and AWS Glue

Configuring AWS Glue Data Catalog as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an AWS Glue Data Catalog Source

Deleting an AWS Glue Data Catalog Source

Source:
<https://docs.dremio.com/current/data-sources/lakehouse-catalogs/open-catalog-external>

Version: current [26.x]

On this page

Open Catalog (External) Enterprise

The Open Catalog (External) source enables you to connect to Open Catalogs deployed in other Dremio instances. Connectivity to the external Open Catalog is achieved using the Iceberg REST API. Once connectivity is established, users can read from and write to external catalogs. Additionally, user impersonation and vended credentials are enabled by default, providing a consistent governance and security experience across your Dremio deployments.

Key use cases for connecting to external Open Catalogs include:

Cross-cluster data federation: Query and join data from multiple Dremio deployments as if they were local tables, enabling unified analytics across geographically distributed or organizationally separate clusters.

Data mesh architecture: Connect domain-specific Dremio clusters (e.g., finance,

marketing, operations) while maintaining data ownership boundaries, allowing controlled cross-domain data access and collaboration.

Hybrid cloud analytics: Access data from on-premises Dremio clusters while running analytics workloads in cloud-based clusters, or vice versa, supporting gradual cloud migration strategies.

Environment promotion workflows: Connect production analytics clusters to development/staging Dremio instances to validate queries, test data products, or promote curated datasets across environments.

Multi-tenant analytics: Enable secure data sharing between different business units or customers, each with their own Dremio cluster, while maintaining isolation and access controls.

When connecting to an external Open Catalog:

Query processing: All local queries run on local engines only. The local Dremio directly reads and writes data from the external catalog's object store but does not delegate processing to the external Dremio cluster. The external Dremio can process queries on its own engines.

User authentication: With impersonation enabled, the local Dremio passes the current user's identity to the external Dremio for privilege checks. User accounts must exist on both clusters with matching identities. Using a shared identity provider (such as LDAP or OIDC) across both clusters simplifies this requirement.

Network requirements: The local Dremio needs direct network access to both the external Dremio catalog and the external catalog's object store (such as S3 or ADLS) to read and write data.

Configure the External Dremio

Log in to the external Dremio as the administrator.

Create a service account that will be used to accept Open Catalog connections.

Create an Inbound Impersonation policy that allows the service user to impersonate the users or groups who will be issuing queries. When creating the impersonation policy, the service account is configured as the `proxy_principal`, and the users or groups submitting data requests are the `target_principals`. The `target_principal` users and groups must match on the local and external Dremio clusters.

Example Inbound Impersonation Policy

```
ALTER SYSTEM SET "exec.impersonation.inbound_policies"='[
  {
    proxy_principals:{
      users:["service-account"]
    },
    target_principals:{
      groups:["external-catalog-users"]
    }
  }
]'
```

Grant the service account the `USAGE` privilege on each folder and `SELECT` on the catalog:

```
`GRANT USAGE ON CATALOG <external-catalog> TO USER <service-account>`
```

```
`GRANT USAGE` on each folder in the catalog `TO USER <service-account>`
```

```
`GRANT SELECT ON CATALOG <external-catalog> TO USER <service-account>`
```

Grant catalog or dataset permissions to the `target_principal` users or groups. The privileges of the `target_principals` will determine whether a data request from that user is granted.

Log in to the external Dremio as the new service account and generate a personal access token (PAT). This PAT will be used to create connections to the external Dremio.

Configure the Local Dremio

On the Datasets page, to the right of **Sources** in the left panel, click [!Add Source icon](#).

In the Add Data Source dialog, under **Lakehouse Catalogs**, select **Open Catalog (External)**.

General

To configure the source connection:

Name: Enter a name for the source. The name you enter must be unique in the organization. Also, consider a name that is easy for users to reference. This name cannot be edited once the source is created. The name cannot exceed 255 characters and must contain only the following characters: `0-9`, `A-Z`, `a-z`, underscore `_`, or hyphen `-`.

Open Catalog Endpoint URL: Specify the Open Catalog endpoint URL of the target Open Catalog. An example is `http://dremio.example.com:8181/api/catalog`.

OAuth Token Endpoint: Specify the OAuth token endpoint of the target Dremio. An example for this endpoint is `http://dremio.example.com:9047/oauth/token`.

PAT Token: Specify the PAT created in the target cluster for the `service-account`. This PAT is used to authenticate to the cluster.

Allow Impersonation: Enabled by default. This setting directs Dremio to execute queries as the user that submits them, utilizing the inbound impersonation policy created on the external Dremio. If user impersonation is disabled, the source credentials will be used to access the catalog.

Storage

Use the **Storage access** field to configure your preferred authentication method. Open Catalog (External) supports two types of credentials for authentication:

Use credential vending (Recommended): Credential vending is a security mechanism where the catalog service issues temporary, scoped access credentials to

the query engine for accessing table storage. The engine is "vended" a temporary credential just in time for the query.

Use master storage credentials: Enter the credentials for accessing all storage URIs within this catalog. If the Iceberg tables' data resides in storage locations other than those listed, Dremio will not be able to access the data.

AWS – Select **AWS** for Amazon S3 and S3-compatible storage. You can refer to the Dremio documentation for connecting to [Amazon S3](#), which is also applicable here. When selecting to assume an IAM role, ensure that the [role policy grants access](#) to the bucket or folder specified by the external catalog.

Azure – Select **Azure** for Azure Blob Storage. You can refer to the Dremio documentation for connecting to [Azure Storage](#), which is also applicable here.

Google Cloud Storage – Select **Google** for Google Cloud Storage (GCS). You can refer to the Dremio documentation for connecting to [GCS](#), which is also applicable here.

The **Connection Properties** are the same as the Advanced Options connection properties from the selected object storage provider above. Refer to the documentation links for your chosen provider (S3, Azure, or GCS) for details on available connection properties and their configuration.

Advanced Options

Cache Options

Enable local caching when possible: Selected by default. Along with asynchronous access for cloud caching, local caching can improve query performance. See [Cloud Columnar Cache](#) for details.

Max percent of total available cache space to use when possible: Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter a percentage in the value field or use the arrows to the far right to adjust the percentage.

Reflection Refresh

You can set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed. See the following options.

Refresh Settings

Never refresh: Select to prevent automatic reflection refresh. The default is to refresh automatically.

Refresh every: How often to refresh reflections, specified in hours, days, or weeks. This option is ignored if **Never refresh** is selected.

Set refresh schedule: Specify the daily or weekly schedule.

Expire Settings

Never expire: Select to prevent reflections from expiring. The default is to expire automatically after the time limit below.

Expire after: The time limit after which reflections expire and are removed from Dremio, specified in hours, days, or weeks. This option is ignored if **Never expire** is selected.

Metadata

Specifying metadata options is handled with the following settings.

Dataset Handling

Remove dataset definitions if the underlying data is unavailable (default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and then replaced with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery:

Fetch every: The refresh interval for fetching top-level source object names, such as databases and tables. Set the time interval using this parameter. You can choose to set the frequency for fetching object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning, such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information:

Fetch mode: You can choose to fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to **Only Queried Datasets**.

Fetch every: You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.

Expire after: You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.

Privileges

You have the option to grant privileges to specific users or roles. See [Access Control](#) for additional information about privileges.

To grant access to a user or role:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Open Catalog (External)

To update an Open Catalog (External) source:

On the Datasets page, under **Sources** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the  at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name.

Click **Save**.

Deleting an Open Catalog (External) Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Open Catalog (External) source:

On the Datasets page, click **Sources > Lakehouse Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

Was this page helpful?

[\[Previous](#)

Lakehouse Catalogs](/current/data-sources/lakehouse-catalogs/)[Next

AWS Glue Data Catalog](/current/data-sources/lakehouse-catalogs/aws-glue-catalog)

Configure the External Dremio

-

Configure the Local Dremio

General

Storage

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an Open Catalog (External)

Deleting an Open Catalog (External) Source

Source:
<https://docs.dremio.com/current/data-sources/warehouse-catalogs/hive/>

Version: current [26.x]

On this page

Hive

This topic describes Hive data source considerations and Dremio configuration.

Dremio and Hive

Dremio supports the following:

Hive 2.1

Hive 3.x

Data Sources

The following data sources are supported:

HDFS

Azure Storage

S3 - See [S3 on Amazon EMR Configuration](#) for more information about S3-backed Hive tables on Amazon EMR.

Hive external tables backed by HBase storage handler

Formats

The following formats are supported:

Apache Avro

Apache Iceberg

Apache Parquet

Delta Lake

ORC

RCFile

SequenceFile

Text, including CSV (Comma-separated values)

In addition, the following interfaces and reading file formats are supported:

Hive table access using Hive's out-of-the-box SerDes interface, as well as custom SerDes or InputFormat/OutputFormat.

Hive-supported reading file format using Hive's own readers even if Dremio does not support them natively.

note

Dremio does *****not***** support Hive views. However, you can create and query views instead.

Hive Configuration

This section provides information about Hive configuration.

Adding additional elements to Hive plugin classpaths

Hive plugins can be extended to use additional resource files and classes. The plugins can be added as either directories or JAR files. Note that any resources that are part of the server's classpath are not exposed to the Hive plugin.

To add additional classpath elements, follow these steps on every node of your Dremio cluster:

Create the following directory:

`<dremio-root>/plugins/connectors/<hive-plugin-id>.d/`

where:

`<dremio-root>` is the root directory of the Dremio instance.

`<hive-plugin-id>` is either of these values:

If you are using Dremio Community/OSS and either Hive 2 or Hive 3: `hive3`

If you are using Dremio Enterprise and either Hive 2 or Hive 3: `hive3-ee`

-

Either place each JAR file in the new directory or add a symlink to each JAR file from the new directory.

Either place a copy of each resource directory in the new directory or add a symlink to each resource directory from the new directory.

Ensure the directory and its contents are readable by the Dremio process user.

Configuration Files

Hive plugins do not use elements present in the main Dremio server classpath. This includes any Hadoop/Hive configuration files such as `core-site.xml` and `hive-site.xml` that the user may have added themselves.

You can add these files to the Hive plugin classpath by following the instructions above.

For example you can create conf files here:

`<dremio-root>/plugins/connectors/**hive3-ee.d**/conf` for the Hive 3 plugin in Enterprise mode.

An easy way to use the same configuration as Dremio is to use a symlink. From ``<dremio-root>``:

Use symlink

```
ln -s conf plugins/connectors/hive3-ee.d/conf
```

Impersonation

note

If you are using Ranger-based authorization for your Hive source, refer to [Disabling Impersonation for Ranger-Based Authorization](#).

To grant the Dremio service user the privilege to connect from any host and to impersonate a user belonging to any group, modify the **core-site.xml** file with the following values:

Grant user impersonation privileges

```
<property>
  <name>hadoop.proxyuser.dremio.hosts</name>
  <value>*</value>
</property>
<property>
  <name>hadoop.proxyuser.dremio.groups</name>
  <value>*</value>
</property>
<property>
  <name>hadoop.proxyuser.dremio.users</name>
  <value>*</value>
</property>
```

To modify the properties to be more restrictive by passing actual hostnames and group names,
modify the **core-site.xml** file with the following values:

Grant more restrictive user impersonation privileges

```
<property>
  <name>hadoop.proxyuser.super.hosts</name>
  <value>10.222.0.0/16,10.113.221.221</value>
</property>
<property>
  <name>hadoop.proxyuser.dremio.users</name>
  <value>user1,user2</value>
</property>
```

Disabling Impersonation for Ranger-Based Authorization

If you are using Ranger-based authorization, we recommend that you disable impersonation for your Hive source:

In the Dremio console, open the **Source Settings** for the Hive source and click **Advanced Options**.

Under **Connection Properties**, add the property `hive.server2.enable.doAs` in the **Name** field and add the setting `false` in the **Value** field.

Click **Save**.

Table Statistics

By default, Dremio utilizes its own estimates for Hive table statistics when planning queries.

However, if you want to use Hive's own statistics, do the following:

Set the `store.hive.use_stats_in_metastore` parameter to true.

Example: `true`: `store.hive.use_stats_in_metastore`

Run the `ANALYZE TABLE COMPUTE STATISTICS` command for relevant Hive tables in Hive.

This step is required so that all of the tables (that Dremio interacts with), have up-to-date statistics.

ANALYZE TABLE COMPUTE STATISTICS command

```
ANALYZE TABLE <Table1> [PARTITION(col1,...)] COMPUTE STATISTICS;
```

Hive Metastores

If you are using a Hive source and an HA metastore (multiple Hive metastores), then you need to specify the following `hive.metastore.uris` parameter and value in the

hive-site.xml file.

Specify hive.metastore.uris

```
<name>hive.metastore.uris</name>
<value>thrift://metastore1:9083,thrift://metastore2:9083</value>
```

Configuring Hive as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Metastores**, select **Hive 2.x** or **Hive 3.x**.

General Options

Name Hive source name. The name cannot include the following special characters: \/, \:, \[, or \].

Connection Hive connection and security

Hive Metastore Host IP address. Example: 123.123.123.123

Port Port number. Default: 9083

Enable SASL Box to enable SASL. If you enable SASL, specify the Hive Kerberos Principal.

Authorization Authorization type for the client.

When adding a new Hive source, you have the following client options for Hive authorization:

Storage Based with User Impersonation A storage-based authorization in the Metastore Server

which is commonly used to add authorization to metastore server API calls.

Dremio utilizes user impersonation to implement Storage Based authorization

When **Allow VDS-based Access Delegation** is enabled (default), the owner of the view is used as the impersonated username.

When **Allow VDS-based Access Delegation** is disabled (unchecked), the query user is used as the impersonated username.

SQL Based **Not Currently Supported**

Ranger Based An Apache Ranger plug-in that provides a security framework for authorization.

Ranger Service Name - This field corresponds to the security profile in Ranger. Example: `hivedev`

Ranger Host URL - This field is the path to the actual Ranger server. Example: `http://yourhostname.com:6080`

Advanced Options

The following options allow you to specify either impersonation users and Hive connection properties.

For example, to add a new Hive source, you can specify a single metastore host by adding a `hive.metastore.uris` parameter and value in the Hive connection properties.

This connection property overrides the value specified in the Hive source.

note

Multiple Hive Metastore Hosts: If you need to specify multiple Hive metastore hosts, update the **hive-site.xml** file.

See Hive Metastores for more information.

Impersonation User Delegation Specifies whether an impersonation username is As is (Default), Lowercase, or Uppercase

Connection Properties Name and value of each Hive connection property.

Credentials Name and hidden value of each Hive connection property for which you want to keep the value secret.

!Dremio Advanced Options

Kerberized Hive

To connect to a Kerberized Hive source, add the following connection property in the Advanced Options:

Property	Description	Value
yarn.resourcemanager.principal	Name of the Kerberos principal for the YARN resource manager.	`<user>/<localhost>@<YOUR-REALM.COM>`

Reflection Refresh

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, or All Datasets. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.
This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.
This mode increases query performance because less work is needed at query time.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Authorization Used when impersonation is enabled.
Specifies the maximum of time that Dremio caches authorization information before expiring.

Expire after - Specifies the expiration time based on minutes, hours, days, or weeks. Default: 1 day

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Hive Source

To update a Hive source:

On the Datasets page, under **Metastores** in the panel on the left, find the name of the

source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Hive as a Source](#).

Click **Save**.

Deleting a Hive Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Hive source, perform these steps:

On the Datasets page, click **Sources > Metastores** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

For More Information

See [Hive Data Types](#) for information about mapping to Dremio data types.

Was this page helpful?

[Previous

Iceberg Catalog](/current/data-sources/lakehouse-catalogs/iceberg-rest-catalog)[Next
Nessie](/current/data-sources/lakehouse-catalogs/nessie)

REST

Dremio and Hive

Data Sources

Formats

-

Hive Configuration

Adding additional elements to Hive plugin classpaths

Configuration Files

Impersonation

Table Statistics

Hive Metastores

Configuring Hive as a Source

General Options

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Hive Source

Deleting a Hive Source

For More Information

Source:
<https://docs.dremio.com/current/data-sources/lakehouse-catalogs/iceberg-rest-catalog/>

Version: current [26.x]

On this page

Iceberg REST Catalog Enterprise

The Iceberg REST Catalog source allows you to connect to your Iceberg Metastores via the Iceberg REST API. This may require configuring specific Advanced Options to set up the correct authentication flows.

Configuring an Iceberg REST Catalog Source

To add an Iceberg REST Catalog source:

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Lakehouse Catalogs**, select **Iceberg REST Catalog Source**.

The New Iceberg REST Catalog Source dialog box appears, which contains the following

tabs:

General: Create a name for your Iceberg REST Catalog source, specify the endpoint URI, and set the authentication Vended credentials (on by default).

Advanced Options: Use catalog properties and credentials to set up storage authentication and authorization.

Reflection Refresh: (Optional) Set a policy to control how often Reflections are refreshed and expired.

Metadata: (Optional) Specify dataset handling and metadata refresh.

Privileges: (Optional) Add privileges for users or roles.

Refer to the following sections for guidance on how to edit each tab.

General

To configure the source connection:

For **Name**, enter a name for the source.

note

The name you enter must be unique in the organization. Also, consider a name that is easy for users to reference. This name cannot be edited once the source is created. The name cannot exceed 255 characters and must contain only the following characters: 0-9, A-Z, a-z, underscore(_), or hyphen (-)

For **Endpoint URI**, specify the catalog service URI.

By default, **Use vended credentials** is turned on. This allows Dremio to connect to the catalog and receive temporary credentials for the underlying storage location. When this setting is enabled, you don't need to add the storage authentication in **Advanced Options**.

note

If you experience errors using vended credentials, please turn the setting off and provide credentials via **Advanced Options** to establish a connection.

(Optional) For **Allowed Namespaces**, add each namespace and check the option if you want to include their whole subtrees. Tables are organized into namespaces, which can be at the top level or nested within one another. Namespace names cannot contain periods or spaces.

Advanced Options

To set the advanced options:

(Optional) For **Catalog Properties** and **Catalog Credentials**, you can manually provide the storage authentication if you choose to not use vended credentials.

Dremio supports Amazon S3 and Azure Storage as object storage services. For acceptable storage authentication configurations, see the following catalog properties

and credentials for each service option.

Amazon S3 Access Key

Type	Name	Value	Description
---	---	---	---
	property		`fs.s3a.aws.credentials.provider`
			`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider` Required field for a Iceberg REST Catalog source
credential		`fs.s3a.access.key` ` <code><your_access_key></code> `	AWS access key ID used by S3A file system. Omit for IAM role-based or provider-based authentication
credential		`fs.s3a.secret.key` ` <code><your_secret_key></code> `	AWS access key used by S3A file system. Omit for IAM role-based or provider-based authentication

Amazon S3 Assumed Role

Type	Name	Value	Description
---	---	---	---
	property		`fs.s3a.assumed.role.arn`
			`arn:aws:iam::*****:role/OrganizationAccountAccessRole` AWS ARN for the role to be assumed
	property		`fs.s3a.aws.credentials.provider`
			`com.dremio.plugins.s3.store.STSCredentialProviderV1` Required field for an Iceberg REST Catalog source
	property		`fs.s3a.assumed.role.credentials.provider`
			`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider` Use only if the credential provider is `AssumedRoleCredentialProvider`; lists credential providers to authenticate with the STS endpoint and retrieve short-lived role credentials

Azure Storage Shared Key

Type	Name	Value	Description
---	---	---	---
credential		`fs.azure.account.key` ` <code><your_account_key></code> `	Storage account key

Under **Cache Options**, review the following table and edit the options to meet your needs.

Cache Options	Description
---	---
Enable local caching when possible	Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.
Max percent of total available cache space to use when possible	Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter in a percentage in the value field or use the arrows to the far right to adjust the percentage.

Reflection Refresh

You can set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed. See the following options.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Set refresh schedule	Specify the daily or weekly schedule.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata

Specifying metadata options is handled with the following settings.

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

Parameter	Description
---	---
Fetch every	You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

Parameter	Description
---	---
Fetch mode	You can choose to fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to Only Queried Datasets .
Fetch every	You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.
Expire after	You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.

Privileges

You have the option to grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges.

To grant access to a user or role:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Iceberg REST Catalog Source

To update an Iceberg REST Catalog:

On the Datasets page, under **Lakehouse Catalogs** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the Source Settings dialog, edit the settings you wish to update. Dremio does not support updating the source name.

Click **Save**.

Deleting an Iceberg REST Catalog Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Iceberg REST Catalog source:

On the Datasets page, click **Sources > Lakehouse Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to

break.

Supported Configurations

The list below contains supported configurations that have been tested with Dremio. The tables outline the parameters needed to connect to the various catalogs. These configurations can be adjusted into REST API calls using the `RESTCATALOG` source type and the `propertyList` and `secretPropertyList` property groups.

All the values below for URI, warehouse, and credentials are example values. These values will need to be changed based on your environment.

Apache Polaris OSS Backed by S3

UI Tab	Field	Value
General	Endpoint URI	`<http://localhost:8181/api/catalog>`
General	Use vended credentials	Unchecked
Advanced Options - Catalog Properties	warehouse	`<polaris_oss_catalog>`
Advanced Options - Catalog Properties	scope	`PRINCIPAL_ROLE:ALL`
Advanced Options - Catalog Properties	fs.s3a.aws.credentials.provider	`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`
Advanced Options - Catalog Credentials	fs.s3a.access.key	`<s3AccessKey>`
Advanced Options - Catalog Credentials	fs.s3a.secret.key	`<s3SecretKey>`
Advanced Options - Catalog Credentials	credential	`<client_id:client_secret>`

Nessie Catalog Backed by S3

UI Tab	Field	Value
General	Endpoint URI	`<http://127.0.0.1:19120/iceberg/>`
General	Use vended credentials	Unchecked
Advanced Options - Catalog Properties	warehouse	`<s3://mybucket/restcatalog/>`
Advanced Options - Catalog Properties	fs.s3a.aws.credentials.provider	`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`
Advanced Options - Catalog Credentials	fs.s3a.access.key	`<s3AccessKey>`
Advanced Options - Catalog Credentials	fs.s3a.secret.key	`<s3SecretKey>`

AWS Glue Iceberg REST Catalog

Please replace `region` uses with a valid AWS region where you are working with the Glue Iceberg REST endpoint (for example, `us-west-2`). You will also need your `AWS account number` and the name of the `Table Bucket` being used.

UI Tab	Field	Value
General	Endpoint URI	`<https://glue.region.amazonaws.com/iceberg>`
General	Use vended credentials	Unchecked
Advanced Options - Catalog Properties	warehouse	`<accountnumber:s3tablescatalog/tablebucketname>`
Advanced Options - Catalog Properties	rest.sigv4-enabled	`true`

Advanced Options - Catalog Properties	rest.signing-name	`glue`
Advanced Options - Catalog Properties	rest.signing-region	`<region>`
Advanced Options - Catalog Properties	fs.s3a.aws.credentials.provider	`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`
Advanced Options - Catalog Properties	dremio.bucket.discovery.enabled	false
Advanced Options - Catalog Properties	dremio.s3.region	`<region>`
Advanced Options - Catalog Properties	fs.s3a.audit.enabled	`false`
Advanced Options - Catalog Properties	fs.s3a.create.file-status-check	false
Advanced Options - Catalog Credentials	rest.access-key-id	`<s3AccessKey>`
Advanced Options - Catalog Credentials	rest.secret-access-key	`<s3SecretKey>`
Advanced Options - Catalog Credentials	fs.s3a.access.key	`<s3AccessKey>`
Advanced Options - Catalog Credentials	fs.s3a.secret.key	`<s3SecretKey>`

S3 Tables Iceberg REST Catalog

Please replace `region` uses with a valid AWS region where you are working with the Glue Iceberg REST endpoint (for example, us-west-2). You will also need your `AWS account number` and the name of the `Table Bucket` being used.

UI Tab	Field	Value
---	---	---
General	Endpoint URI	`<https://s3tables.region.amazonaws.com/iceberg>`
General	Use vended credentials	Unchecked
Advanced Options - Catalog Properties	warehouse	`<arn:aws:s3tables:region:accountnumber:bucket/tablebucketname>`
Advanced Options - Catalog Properties	rest.sigv4-enabled	`true`
Advanced Options - Catalog Properties	rest.signing-name	`s3tables`
Advanced Options - Catalog Properties	rest.signing-region	`<region>`
Advanced Options - Catalog Properties	fs.s3a.aws.credentials.provider	`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`
Advanced Options - Catalog Properties	dremio.bucket.discovery.enabled	false
Advanced Options - Catalog Properties	dremio.s3.region	`<region>`
Advanced Options - Catalog Properties	fs.s3a.audit.enabled	`false`
Advanced Options - Catalog Properties	fs.s3a.create.file-status-check	`false`
Advanced Options - Catalog Credentials	rest.access-key-id	`<s3AccessKey>`
Advanced Options - Catalog Credentials	rest.secret-access-key	`<s3SecretKey>`
Advanced Options - Catalog Credentials	fs.s3a.access.key	`<s3AccessKey>`
Advanced Options - Catalog Credentials	fs.s3a.secret.key	`<s3SecretKey>`

Tableflow Catalog backed by AWS

Note that namespaces for the Tableflow Catalog are the Kafka clusters within your environment.

UI Tab	Field	Value
---	---	---
General	Endpoint URI	`<https://tableflow.us-west-2.aws.confluent.cloud/iceberg/catalog/organizations/f140b886-a3e9-4e1d-ba9d-5b96b8bf4ea8/environments/env-7kn93o>`
General	Allowed Namespaces	`<kafkaClusterID>`
General	Allowed Namespaces include their whole subtrees	Unchecked
General	Use vended credentials	Checked
Advanced Options - Catalog Credentials	credential	`<api_key:secret_key>`

Was this page helpful?

[Previous

Unity Catalog](/current/data-sources/lakehouse-catalogs/unity)[Next

Hive](/current/data-sources/lakehouse-catalogs/hive)

Configuring an Iceberg REST Catalog Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an Iceberg REST Catalog Source

Deleting an Iceberg REST Catalog Source

Supported Configurations

Apache Polaris OSS Backed by S3

Nessie Catalog Backed by S3

AWS Glue Iceberg REST Catalog

S3 Tables Iceberg REST Catalog

Tableflow Catalog backed by AWS

Source:
<https://docs.dremio.com/current/data-sources/lakehouse-catalogs/nessie/>

Version: current [26.x]

On this page

Nessie

Nessie is an intelligent metastore and catalog for Apache Iceberg. It provides a modern alternative to Hive Metastore for Iceberg tables and views and provides many advanced features for more effective data lakes. These features include:

Adding or changing data on a versioned branch, testing that branch for quality, and merging the changes to general user availability, all within the same data lake and without impacting production data.

Creating specialized versions of data for specific use cases.

Atomically updating many tables, with many changes, thus eliminating data

inconsistencies and aberrations in the middle of a change sequence.

Concepts

Architecture

The Nessie service is a lightweight Java-based REST API server. Nessie uses configurable authentication and a configurable backend datastore (which currently supports multiple database types). This architecture allows Nessie to run in one or more Docker instances according to capacity requirements. The [Nessie Helm chart](#) deploys the front end load balancer and assists with other details such as the configuration HTTPS. The Nessie JAR file can be deployed when a single Nessie instance is required for test purposes, or for a local development or test environment.

!Nessie diagram showing how Dremio communicates with a standalone Nessie server.

Objects in Nessie

When working with a Nessie source, you work in or with the following objects:

Branch: A named reference and a movable pointer to a commit.

Folders: Used to help you organize your tables in a Nessie source.

Tables: Contains the data from your source, formatted as rows and columns. A table can be modified by query engines that connect to your Nessie source.

Views: A virtual table, created by running SQL statements or functions on a table or another view.

You can create and store Apache Iceberg tables and views in the Nessie catalog. No other file or source types can be stored in the Nessie catalog.

Git-like Data Management

Nessie is a native [Apache Iceberg catalog](#) that provides Git-like data management. As a result, data engineering teams can use commits, branches, and tags to be able to experiment on Apache Iceberg tables.

Commit: A transaction affecting one or more tables or views. It may take place over a short or long period of time. Examples include:

Updating a table using Dremio (`INSERT`, `UPDATE`, `DELETE`, `MERGE`, `TRUNCATE`) or another engine such as Spark

Updating a view or the definition of a view

Updating the schema of a table via SQL (`ALTER TABLE`) or Spark

Branch: A movable pointer to a commit. Every time you commit, the branch pointer moves forward automatically. Branches can be merged via a commit.

Tag: A named commit. You can tag a commit with a specific name so that users can

refer to it without specifying a commit hash.

These capabilities enable a variety of use cases such as:

Multi-statement transactions: With branches, data is updated in isolation and changes are merged atomically. The updates can be performed through a single engine (for example, SQL DML statements in Dremio) or through multiple engines (for example, ingest data in Spark and delete a record in Dremio), and may span any period of time and any number of users.

Experimentation: Experimenting on the live lakehouse risks exposing incorrect or inconsistent data to other users. Instead, you can easily create a sandbox branch and experiment there. Because the data is not duplicated, there is no cost to creating a sandbox. And when you are done, the branch can be either deleted or your changes can be merged into the main branch.

Reproducibility: The ability to retrain machine learning models and BI dashboards based on historical data is important for reproducible research and regulation. Nessie enables any engine to access previous versions of the lakehouse by referencing a specific commit, tag (a named commit), or timestamp.

Governance: Nessie provides a user interface familiar to users of GitHub and GitLab that makes it easy to see every commit in every branch, so that you don't have to wonder who updated or deleted a table, or where a table originated.

The following illustration shows an example of a new branch that is forked from the main branch, then merged back atomically after multiple commits:

[!Example of Git branching.](#)

Prerequisites

Dremio supports Nessie version 0.59.0 and later. If you have not yet set up a Nessie server and connected it with your dataset, you can choose to either set up a server in a [fast-start Docker image](#) or with [secure HTTPS transport in Minikube](#).

When using Nessie as a source, Dremio can connect to Amazon S3 buckets, Azure Storage, Google Cloud Storage (GCS), or S3-compatible storage providers like MinIO and Dell ECS. Read [Storage](#) for details about the required credentials for connecting to each storage provider.

Configuring Nessie as a Source

To add a Nessie source to your project:

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Nessie Catalogs**, select **Nessie**.

The New Nessie Source dialog box appears, which contains the following sections:

General: Create a name for your Nessie source, specify the endpoint URL, and set the authentication type. The name cannot include the following special characters: ``/`, `:``, `[`, or `]``.

Storage: Set the storage option by setting up the authentication type and the connection properties.

Advanced Options: (Optional) Use the default settings or, optionally, configure access preferences and cache options.

Privileges: (Optional) Add privileges for users or roles.

Refer to the following for guidance on how to edit each section.

General

This tab provides options for configuring connections to a Nessie source.

In the **Name** field, enter a name.

note

The name you enter must be unique in the organization. Also, consider a name that is easy for users to reference. This name cannot be edited once the source is created. The name cannot exceed 255 characters and must contain only the following characters: 0-9, A-Z, a-z, underscore(_), or hyphen (-).

In the **Nessie endpoint URL** field, specify the IP address and port that you have set up for your Nessie server (e.g., `https://localhost:19120/api/v2`). For more information, see [Project Nessie Configuration](#).

Under **Nessie authentication type**, select either **None** or **Bearer**:

None: The Nessie server does not require authentication.

Bearer: Set authentication using an OpenID bearer token. For more information about setting up this type of authentication, see [Project Nessie's Authentication page](#). Then, choose a method for providing the password from the dropdown menu:

Dremio: Provide the bearer token in plain text. Dremio stores the password.

Azure Key Vault: Store the bearer token securely using URI format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the bearer token, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Choose the HashiCorp secrets engine you're using from the dropdown menu and enter the secret reference for the bearer token in the correct format in the provided field.

Storage

Nessie sources can use Amazon S3 buckets (AWS), Azure Storage (Azure), Google Cloud Storage (Google), or S3-compatible storage providers like MinIO and Dell ECS as storage.

AWS

Azure

Google

To connect an Amazon S3 bucket or a S3-compatible storage provider to the Nessie source, select the **AWS** storage provider option.

S3 Storage

In the field under **AWS root path**, provide the root path of the S3 bucket to use. We recommend that you have either a dedicated S3 bucket or a dedicated folder in which to store Nessie objects.

Authentication

Under **Authentication method**, choose the method you want to use to authenticate to Amazon S3.

AWS Access Key:

In the field under **AWS access key**, provide the access key for the Amazon S3 account.

Under **AWS access secret**, use the dropdown menu to choose a method for providing the access secret for the Amazon S3 account:

Dremio: Provide the access secret in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored access secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the access secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the access secret reference in the required format.

In the field under **IAM role to assume**, provide the ARN of the IAM role.

EC2 Metadata: In the field under **IAM role to assume**, provide the ARN of an IAM role with privileges on the S3 bucket. This role could be attached to the EC2 instance or to an IAM role to assume for connecting to the S3 bucket. In either case, the role must provide privileges to use the S3 bucket.

AWS Profile: In the field under **AWS profile (optional)**, provide the AWS Profile name. If you leave the field blank, Dremio uses the default AWS Profile.

No Authentication: Select this option if no credentials are required because you are connecting the Nessie source to a public Amazon S3 bucket.

If you are connecting to S3-compatible storage like MinIO or Dell ECS, choose **AWS access key** for authentication and provide the access key and secret.

Other: Connection Properties

Provide the custom key-value pairs for the connection relevant to the source.

(Optional) If you are connecting to S3 storage, complete the following:

Click **Add Property**.

For **Name**, provide a connection property.

For **Value**, provide the corresponding value for the connection property.

If you are connecting to S3-compatible storage like MinIO or Dell ECS, complete the following:

Add ``fs.s3a.path.style.access`` and set the value to ``true``. This setting ensures that the request path is created correctly when using IP addresses or hostnames as the endpoint.

Add ``fs.s3a.endpoint`` property and its corresponding server endpoint value (IP address). The endpoint value cannot contain the ``http(s)://`` prefix nor can it start with the string ``s3``. For example, if the endpoint is ``http://123.1.2.3:9000``, the value is ``123.1.2.3:9000``.

Add ``dremio.s3.compat`` and set the value to ``true``.

Other: Encrypt connection

Optional: To secure the connections between the Amazon S3 bucket and Dremio, select the **Encrypt connection** checkbox.

To save the configuration, click **Save**. To configure additional settings, proceed to [Advanced Options](#).

To connect Azure Storage to the Nessie source, select the **Azure** storage provider option.

Azure Storage

In the field under **Storage Account Name**, provide the name of the Azure Storage account to use.

In the field under **Azure root path**, provide the path in your Azure Storage account to the write location that Dremio should use for Iceberg metadata and data. The root path includes the name of the Azure Storage container, followed by the names of any folders (for example, ``/containername/optional/folder/path``).

Azure Authentication

Under **Authentication method**, choose whether you want to authenticate to Azure Storage with a shared access key or Microsoft Entra ID.

Shared access key: Use the dropdown menu to choose a method for providing the shared access key for the Azure Storage account:

Dremio: Provide the access key in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored access key using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the access key, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the access key reference in the required format.

Microsoft Entra ID:

In the field under **Application ID**, provide the ID for the application (client) in Azure.

Under **Client secret**, use the dropdown menu to choose a method for providing the client secret for the Azure Storage account:

Dremio: Provide the client secret in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored client secret using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the client secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the client secret reference in the required format.

In the field under **OAuth 2.0 token endpoint**, provide the OAuth 2.0 token endpoint (v1.0), including the tenant ID, that the application uses to get an access token or a refresh token.

Other: Connection Properties (Optional)

Provide the custom key-value pairs for the connection relevant to the source.

Click **Add Property**.

For **Name**, provide a connection property.

For **Value**, provide the corresponding value for the connection property.

Other: Encrypt connection

Optional: To secure the connections between Azure Storage and Dremio, select the **Encrypt connection** checkbox.

To save the configuration, click **Save**. To configure additional settings, proceed to Advanced Options.

To connect Google Cloud Storage (GCS) to the Nessie source, select the **Google** storage provider option.

GCS Storage

In the field under **Google Project ID**, provide the ID for your GCS project. You can find the ID in the **Project info** pane at the top-left of your screen on the GCS Home page.

In the field under **Google root path**, provide the path for the GCS source that Dremio should use for Iceberg metadata and data.

GCS Authentication

Under **Authentication method**, choose whether you want to authenticate to GCS with a service account key or by automatic/service account.

Service Account Keys:

In the field under **Client Email**, provide the email address associated with the GCS service account.

In the field under **Client ID**, provide the client ID for your GCS key pair.

In the field under **Private Key ID**, provide the key ID for your GCS key pair.

Under **Private Key**, use the dropdown menu to choose a method for providing the private key for your GCS key pair:

Dremio: Provide the private key in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored private key using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the private key, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the private key secret reference in the required format.

Automatic/Service Account: If you are running Dremio on a Google Compute instance, Dremio uses the active service account for your instance and does not require any additional information to integrate with your data.

Other: Connection Properties (Optional)

Provide the custom key-value pairs for the connection relevant to the source.

Click **Add Property**.

For **Name**, provide a connection property.

For **Value**, provide the corresponding value for the connection property.

Other: Encrypt connection

Optional: To secure the connections between GCS and Dremio, select the **Encrypt connection** checkbox.

To save the configuration, click **Save**. To configure additional settings, proceed to [Advanced Options](#).

Advanced Options

Click **Advanced Options** in the left menu sidebar.

Under Cache Options, review the following table and edit the options to meet your needs.

Cache Options	Description
--- ---	
Enable local caching when possible	Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.
Max percent of total available cache space to use when possible	Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter in a percentage in the value field or use the arrows to the far right to adjust the percentage.

Reflection Refresh

The **Reflection Refresh** section allows you to set a schedule for refreshing all of the Reflections that are defined on tables in the catalog. You can override this schedule on individual tables in different branches. This section also lets you specify how long all Reflections in the catalog exist until they expire. Again, you can override this setting on individual tables in different branches.

To learn more, see [Refreshing Reflections](#) and [Setting the Expiration Policy for Reflections](#).

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

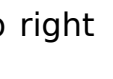
Click **Save** after setting the configuration.

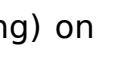
At this point, a connection with the Nessie server is attempted. If a connection cannot be made, report the issue to the Project Nessie community's [Zulip channel](#). You can also file a ticket on the Project Nessie community's [GitHub page](#).

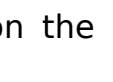
Retrieving a Table Definition

You can retrieve the table definition for Nessie tables if you have the ``SELECT`` privilege on the table. Because tables cannot be modified, you can't make edits to the table definition but you can retrieve the definition to understand where the table was derived from and to use it as a template for creating new views.

To see a table definition on the Datasets page, choose any one of these options:

Hover over the table name and click  in the top right corner of the metadata card.

Hover over the line containing the table and click  on the right.

Hover over the line containing the dataset, click  on the right, and select **Go to Table**.

The table definition opens in the SQL editor.

If you want to use this table definition to create a view, see [Creating a View](#).

tip

If you have the ``SELECT`` privilege on a Nessie table, you can run ``SHOW CREATE TABLE <table_name>`` in the SQL editor to see the table definition. See [SHOW CREATE TABLE](#).

Updating a Nessie Source

To update a Nessie source:

On the Datasets page, under **Nessie Catalogs** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the  at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Nessie as a Source](#).

Click **Save**.

Deleting a Nessie Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Nessie source, perform these steps:

On the Datasets page, click **Sources > Nessie Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and

right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Limitations

Changes to tables and views that are in Nessie sources are not logged. Nessie sources do not have audit logs.

DX-64988

The [Catalog API](#) is unable to retrieve or manage Nessie sources.

DX-64994

Dremio does not support moving, copying, or renaming tables and views in Nessie sources or removing the format from tables in Nessie sources.

Was this page helpful?

[\[Previous](#)

[Hive\]\(/current/data-sources/lakehouse-catalogs/hive\)\[Next](#)

[Object Storage\]\(/current/data-sources/object/\)](#)

[Concepts](#)

[Architecture](#)

[Objects in Nessie](#)

[Git-like Data Management](#)

[Prerequisites](#)

[Configuring Nessie as a Source](#)

[General](#)

[Storage](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Privileges](#)

[Retrieving a Table Definition](#)

-

Updating a Nessie Source

Deleting a Nessie Source

Limitations

Source:
<https://docs.dremio.com/current/data-sources/lakehouse-catalogs/snowflake-open/>

Version: current [26.x]

On this page

Snowflake Open Catalog Enterprise

Dremio supports Snowflake Open Catalog as an Iceberg catalog source. With this source connector, you can connect and read from internal and external Snowflake Open Catalogs and write to External Snowflake Open Catalogs.

Prerequisites

You will need the catalog **Service URI**, **Client ID**, and **Client Secret** from the Snowflake setup. For a walkthrough of the Snowflake setup, please refer to [Query a table in Snowflake Open Catalog using a third-party engine](#).

Configuring Snowflake Open Catalog as a Source

To add a Snowflake Open Catalog source:

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Lakehouse Catalogs**, select **Snowflake Open Catalog**.

The New Snowflake Open Catalog dialog box appears, which contains the following tabs:

General: Create a name for your Snowflake Open Catalog source, specify the endpoint URI and Snowflake Open Catalog, and set the authentication.

Advanced Options: Use catalog properties and credentials to set up storage authentication and authorization.

Reflection Refresh: (Optional) Set a policy to control how often Reflections are refreshed and expired.

Metadata: (Optional) Specify dataset handling and metadata refresh.

Privileges: (Optional) Add privileges for users or roles.

Refer to the following sections for guidance on how to edit each tab.

General

To configure the source connection:

For **Name**, enter a name for the source.

note

The name you enter must be unique in the organization. Also, consider a name that is easy for users to reference. This name cannot be edited once the source is created. The name cannot exceed 255 characters and must contain only the following characters: 0-9, A-Z, a-z, underscore(_), or hyphen (-)

Enter the name of the Snowflake Open Catalog.

For **Endpoint URI**, specify the catalog service URI.

In the Authentication section, use the **Client ID** and **Client Secret** created during the configuration of a service connection for the Snowflake Open Catalog.

By default, `Use vended credentials` is on. This allows Dremio to connect to the catalog and receive temporary credentials to the underlying storage location. If this is enabled, there is no need to add the storage authentication in Advanced Options.

(Optional) For **Allowed Namespaces**, add each namespace and check the option if you want to include their whole subtrees. Tables are organized into namespaces, which can be at the top level or nested within one another. Namespace names cannot contain periods or spaces.

Advanced Options

To set the advanced options:

(Optional) For **Catalog Properties** and **Catalog Credentials**, you can manually provide the storage authentication if you choose to not use vended credentials.

Dremio supports Amazon S3, Azure Storage, and Google Cloud Storage (GCS) as object storage services. For acceptable storage authentication configurations, see the following catalog properties and credentials for each service option.

Amazon S3 Access Key

Type	Name	Value	Description
---	---	---	---
	property	`fs.s3a.aws.credentials.provider`	
		`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`	Required value for a Snowflake Open Catalog source
credential	`fs.s3a.access.key`	`<your_access_key>`	AWS access key ID used by S3A file system
credential	`fs.s3a.secret.key`	`<your_secret_key>`	AWS secret key used by S3A file system

Amazon S3 Assumed Role

Type	Name	Value	Description
---	---	---	---

Type	Name	Value	Description
property	<code>fs.s3a.assumed.role.arn</code>	<code>arn:aws:iam::*****:role/OrganizationAccountAccessRole</code>	AWS ARN for the role to be assumed
property	<code>fs.s3a.aws.credentials.provider</code>	<code>com.dremio.plugins.s3.store.STSCredentialProviderV1</code>	Required value for a Snowflake Open Catalog source
property	<code>fs.s3a.assumed.role.credentials.provider</code>	<code>org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider</code>	Use only if the credential provider is <code>AssumedRoleCredentialProvider</code> ; lists credential providers to authenticate with the STS endpoint and retrieve short-lived role credentials
credential	<code>fs.s3a.access.key</code>	<code><your_access_key></code>	AWS access key ID used by S3A file system
credential	<code>fs.s3a.secret.key</code>	<code><your_secret_key></code>	AWS secret key used by S3A file system

Azure Storage with Microsoft Entra ID

Type	Name	Value	Description
property	<code>fs.azure.account.auth.type</code>	<code>OAuth</code>	
property	<code>fs.azure.account.oauth2.client.id</code>	<code><your_client_ID></code>	Client ID from App Registration within Azure Portal
property	<code>fs.azure.account.oauth2.client.endpoint</code>	<code>https://login.microsoftonline.com/<ENTRA ID>/oauth2/token</code>	Microsoft Entra ID from Azure Portal
credential	<code>fs.azure.account.oauth2.client.secret</code>	<code><your_client_secret></code>	Client secret from App Registration within Azure Portal

Azure Storage Shared Key

Type	Name	Value	Description
credential	<code>fs.azure.account.key</code>	<code><your_account_key></code>	Storage account key

Google Cloud Storage (GCS) Using Default Credentials

Type	Name	Value	Description
property	<code>dremio.gcs.use_keyfile</code>	<code>false</code>	Required value for a Snowflake Open Catalog source

Google Cloud Storage (GCS) Using KeyFile

Type	Name	Value	Description
property	<code>dremio.gcs.clientId</code>	<code><your_client_ID></code>	Client ID from GCS
property	<code>dremio.gcs.projectId</code>	<code><your_project_ID></code>	Project ID from GCS
property	<code>dremio.gcs.clientEmail</code>	<code><your_client_email></code>	Client email from GCS
property	<code>dremio.gcs.privateKeyId</code>	<code><your_private_key_ID></code>	Private key ID from GCS
property	<code>dremio.gcs.use_keyfile</code>	<code>true</code>	Required value for a Snowflake Open Catalog source
credential	<code>dremio.gcs.privateKey</code>	<code><your_private_key></code>	Private key from GCS

Under **Cache Options**, review the following table and edit the options to meet your needs.

Cache Options	Description
---	---
Enable local caching when possible	Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.
Max percent of total available cache space to use when possible	Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter in a percentage in the value field or use the arrows to the far right to adjust the percentage.

Reflection Refresh

You can set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed. See the following options.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Set refresh schedule	Specify the daily or weekly schedule.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata

Specifying metadata options is handled with the following settings.

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

| Parameter | Description |

| --- | --- |

| **Fetch every** | You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour. |

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

| Parameter | Description |

| --- | --- |

| **Fetch mode** | You can choose to fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to **Only Queried Datasets**. |

| **Fetch every** | You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour. |

| **Expire after** | You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours. |

Privileges

You have the option to grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges.

To grant access to a user or role:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Snowflake Open Catalog Source

To update a Snowflake Open Catalog source:

On the Datasets page, under **Lakehouse Catalogs** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Snowflake Open Catalog as a Source](#).

Click **Save**.

Deleting a Snowflake Open Catalog Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Snowflake Open Catalog source:

On the Datasets page, click **Sources > Lakehouse Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Additional Resources

Learn more about Polaris by enrolling in the [Apache Polaris course in Dremio University](#).

Was this page helpful?

[Previous

Microsoft OneLake](/current/data-sources/lakehouse-catalogs/onelake)[Next

Unity Catalog](/current/data-sources/lakehouse-catalogs/unity)

Prerequisites

Configuring Snowflake Open Catalog as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Snowflake Open Catalog Source

Deleting a Snowflake Open Catalog Source

Additional Resources

Source:
<https://docs.dremio.com/current/data-sources/l>

lakehouse-catalogs/unity/

Version: current [26.x]

On this page

Unity Catalog Enterprise

Unity Catalog provides a metastore for Delta tables within the Databricks ecosystem, and Dremio supports these Delta Lake Universal Format (UniForm) tables in Unity Catalog as a Dremio data source.

UniForm Iceberg

UniForm is an Iceberg metadata layer that provides a read-only interoperability layer for Iceberg clients. To query Delta tables, you must use UniForm to read Delta tables with Iceberg clients. For guidance, see [Enable UniForm Iceberg](#) in the Databricks documentation. The minReaderVersion of UniForm required is 2, as noted in [Features by protocol version](#).

For the limitations of UniForm tables, see [Limitations](#) in the Databricks documentation.

Configuring Unity Catalog as a Source

To add a Unity Catalog source:

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Lakehouse Catalogs**, select **Unity Catalog**.

General

To configure the source connection:

Name: Enter a name for the source. The name you enter must be unique in the organization. Also, consider a name that is easy for users to reference. This name cannot be edited once the source is created. The name cannot exceed 255 characters and must contain only the following characters: 0-9, A-Z, a-z, underscore(_), or hyphen (-)

Endpoint URI: Specify the catalog service URI. For more information on how to find your Unity Catalog URI, see [Read using the Unity Catalog Iceberg catalog endpoint](#) for more details.

Unity Catalog, enter the catalog name.

Authentication

Select an authentication option:

-

Databricks Personal Access Token (PAT): Depending on your deployment, choose from the following options to create a PAT:

AWS-based Unity deployment: See [Databricks personal access tokens for service principals](#) to create a PAT.

Azure Databricks-based Unity deployment: See [Azure Databricks personal access tokens for service principals](#) to create a PAT.

Select a method to store the PAT:

Dremio: Provide the PAT in plain text. Dremio stores the PAT.

Azure Key Vault: Provide the URI for your stored PAT using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the PAT, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the PAT reference in the required format.

Microsoft Entra ID: To register a Microsoft Entra ID application and obtain the required IDs and client secret, see [How to register an app in Microsoft Entra ID](#), then complete the Dremio configuration:

Application ID: Enter the application (client) ID

OAuth 2.0 Token Endpoint: The OAuth 2.0 token endpoint that the application uses to get an access token

Application Secret:

Dremio: Provide the application secret in plain text. Dremio stores the application secret.

Azure Key Vault: Provide the URI for your stored application secret securely using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the application secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine using from the dropdown and enter the secret reference for the application secret in the required format.

By default, **Use vendored credentials** is turned on. This allows Dremio to connect to the catalog and receive temporary credentials for the underlying storage location. When this option is enabled, you don't need to add the storage authentication in **Advanced Options**.

Allowed Namespaces

For **Allowed Namespaces**, optionally add each namespace and check the option if you want to include their whole subtrees. Tables are organized into namespaces, which can

be at the top level or nested within one another. Namespace names cannot contain periods or spaces.

Advanced Options

To set the advanced options:

(Optional) For **Catalog Properties** and **Catalog Credentials**, you must manually provide the storage authentication if you choose to not use vended credentials.

Dremio supports Amazon S3, Azure Storage, and Google Cloud Storage (GCS) as object storage services. For acceptable storage authentication configurations, see the following catalog properties and credentials for each service option.

Amazon S3 Access Key

Type	Name	Value	Description
---	---	---	
	property		<code>`fs.s3a.aws.credentials.provider`</code>
		<code>`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`</code>	Required field for a Unity Catalog source
credential	<code>`fs.s3a.access.key`</code>	<code>`<your_access_key>`</code>	AWS access key ID used by S3A file system. Omit for IAM role-based or provider-based authentication
credential	<code>`fs.s3a.secret.key`</code>	<code>`<your_secret_key>`</code>	AWS access key used by S3A file system. Omit for IAM role-based or provider-based authentication

Amazon S3 Assumed Role

Type	Name	Value	Description
---	---	---	
	property		<code>`fs.s3a.assumed.role.arn`</code>
		<code>`arn:aws:iam::*****:role/OrganizationAccountAccessRole`</code>	AWS ARN for the role to be assumed
	property		<code>`fs.s3a.aws.credentials.provider`</code>
		<code>`com.dremio.plugins.s3.store.STSCredentialProviderV1`</code>	Required field for a Unity Catalog source
	property		<code>`fs.s3a.assumed.role.credentials.provider`</code>
		<code>`org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider`</code>	Use only if the credential provider is <code>`AssumedRoleCredentialProvider`</code> ; lists credential providers to authenticate with the STS endpoint and retrieve short-lived role credentials

Azure Storage Shared Key

Type	Name	Value	Description
---	---	---	
credential	<code>`fs.azure.account.key`</code>	<code>`<your_account_key>`</code>	Storage account key

Google Cloud Storage (GCS) Using KeyFile

Type	Name	Value	Description
---	---	---	
	property		<code>`fs.AbstractFileSystem.gs.impl`</code>
		<code>`com.google.cloud.hadoop.fs.gcs.GoogleHadoopFS`</code>	Required field for a Unity Catalog source

property	<code>`fs.gs.auth.service.account.enable`</code>	<code>`true`</code>	Required field for a Unity Catalog source
property	<code>`fs.gs.impl`</code>	<code>`com.google.cloud.hadoop.fs.gcs.GoogleHadoopFileSystem`</code>	Required field for a Unity Catalog source
property	<code>`fs.gs.bucket`</code>	<code>`<your_bucket>`</code>	Bucket where your data is stored for Unity Catalog in GCS
property	<code>`fs.gs.project.id`</code>	<code>`<your_project_ID>`</code>	Project ID from GCS
property	<code>`fs.gs.auth.service.account.email`</code>	<code>`<your_client_email>`</code>	Client email from GCS
property	<code>`fs.gs.auth.service.account.private.key.id`</code>	<code>`<your_private_key_id>`</code>	Private key ID from GCS
property	<code>`dremio.gcs.use_keyfile`</code>	<code>`true`</code>	Required field for a Unity Catalog source
credential	<code>`fs.gs.auth.service.account.private.key`</code>	<code>`<your_private_key>`</code>	Private key from GCS

Under **Cache Options**, review the following table and edit the options to meet your needs.

Cache Options	Description
---	---
Enable local caching when possible	Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.
Max percent of total available cache space to use when possible	Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter in a percentage in the value field or use the arrows to the far right to adjust the percentage.

Reflection Refresh

You can set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed. See the following options.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Set refresh schedule	Specify the daily or weekly schedule.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata

Specifying metadata options is handled with the following settings.

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

| Parameter | Description |

| --- | --- |

| **Fetch every** | You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour. |

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

| Parameter | Description |

| --- | --- |

| **Fetch mode** | You can choose to fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to **Only Queried Datasets**. |

| **Fetch every** | You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour. |

| **Expire after** | You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours. |

Privileges

You have the option to grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges.

To grant access to a user or role:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Unity Catalog Source

To update a Unity Catalog source:

On the Datasets page, under **Lakehouse Catalogs** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name, and then click the [!The Settings icon](#) at the top-right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Unity Catalog as a Source](#).

Click **Save**.

Deleting a Unity Catalog Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Unity Catalog source:

On the Datasets page, click **Sources > Lakehouse Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[Previous

Snowflake Catalog](/current/data-sources/lakehouse-catalogs/snowflake-open)[Next

Open

Iceberg REST Catalog](/current/data-sources/lakehouse-catalogs/iceberg-rest-catalog)

UniForm Iceberg

Configuring Unity Catalog as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Unity Catalog Source

Deleting a Unity Catalog Source

Source:
<https://docs.dremio.com/current/data-sources/lakehouse-catalogs/onelake>

Version: current [26.x]

On this page

Microsoft OneLake Enterprise

Microsoft OneLake is a single, unified, logical data lake that comes automatically with every Microsoft Fabric tenant. It is built on top of Azure Data Lake Storage (ADLS) Gen2. All Fabric data items, such as data warehouses and lakehouses, store their data automatically in OneLake.

Dremio creates connections to Microsoft OneLake using its Iceberg REST Catalog connector.

Configure a Microsoft OneLake Source

To add a Microsoft OneLake source connection:

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Source dialog, under **Lakehouse Catalogs**, select **Iceberg REST Catalog Source**.

General

To configure the source connection:

For **Name**, enter a name for the source. The name you enter must be unique in the organization. Also, consider a name that is easy for users to reference. This name cannot be edited once the source is created. The name cannot exceed 255 characters and must contain only the following characters: 0-9, A-Z, a-z, underscore (_), or hyphen (-).

For **Endpoint URI**, specify the catalog service URI as ``https://onelake.table.fabric.microsoft.com/iceberg``.

Clear **Use vended credentials**.

(Optional) For **Allowed Namespaces**, add each namespace and check the option if you want to include their entire subtrees. Tables are organized into namespaces, which can be at the top level or nested within one another. Namespace names cannot contain periods or spaces.

Advanced Options

Replace the placeholders inside ``<...>`` with your respective values. For example, a warehouse value could be ``icy/icelake.Lakehouse``. The ``fs.*`` properties are used to establish connections to the storage underlying your catalog in OneLake.

Add the following **Catalog Properties** with their associated values.

`rest.auth.type` : `oauth2``

`oauth2-server-uri` : `https://login.microsoftonline.com/<tenant_id>/oauth2/v2.0/token``

`scope` : `https://storage.azure.com/.default``

`warehouse` : `<catalog>``

`fs.azure.endpoint` : `dfs.fabric.microsoft.com``

`fs.azure.account.auth.type` : `OAuth``

`fs.azure.account.oauth2.client.endpoint` :
`https://login.microsoftonline.com/<tenant_id>/oauth2/v2.0/token``

`fs.azure.account.oauth2.client.id` : `<oauth_client_id>``

Add the following **Catalog Credentials** with their associated values:

`fs.azure.account.oauth2.client.secret` : `<oauth_client_secret>``

`credential` : `<oauth_client_id:oauth_client_secret>``

Under **Cache Options**, review the following table and edit the options to meet your needs.

Enable local caching when possible – Selected by default. Along with asynchronous access for cloud caching, local caching can improve query performance. See [Cloud Columnar Cache](#) for details.

Max percent of total available cache space to use when possible – Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space available on the mount point provided for caching. You can either manually enter a percentage in the value field or use the arrows to the far right to adjust the percentage.

Reflection Refresh

You can set the policy that controls how often reflections are scheduled to be refreshed automatically, as well as the time limit after which reflections expire and are removed.

See the following options:

Option	Description
Never refresh	Select to prevent automatic reflection refresh. The default is to automatically refresh.
Refresh every	How often to refresh reflections, specified in hours, days, or weeks. This option is ignored if Never refresh is selected.
Set refresh schedule	Specify the daily or weekly schedule.
Never expire	Select to prevent reflections from expiring. The default is to automatically expire after the time limit below.
Expire after	The time limit after which reflections expire and are removed from Dremio, specified in hours, days, or weeks. This option is ignored if Never expire is selected.

Metadata

Metadata options are configured using the following settings.

Dataset Handling

Remove dataset definitions if underlying data is unavailable (default).

If this box is *not* checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

Parameter	Description
Fetch every	You can set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning, such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

Parameter	Description
Fetch mode	You can fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to Only Queried Datasets .
Fetch every	You can set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.
Expire after	You can set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.

Privileges

You have the option to grant privileges to specific users or roles. See [Access Control](#) for additional information about privileges.

To grant access to a user or role:

For **Privileges**, enter the user name or role name to which you want to grant access and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Update a Microsoft OneLake Source

To update a Microsoft OneLake source connection:

On the Datasets page, under **Lakehouse Catalogs** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the Source Settings dialog, edit the settings you want to update. Dremio does not support updating the source name.

Click **Save**.

Delete a Microsoft OneLake Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Microsoft OneLake source:

On the Datasets page, click **Sources > Lakehouse Catalogs** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[Previous

AWS

Glue

Data

Catalog][(/current/data-sources/lakehouse-catalogs/aws-glue-catalog)][Next

Snowflake Open Catalog][(/current/data-sources/lakehouse-catalogs/snowflake-open)

Configure a Microsoft OneLake Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Update a Microsoft OneLake Source

Delete a Microsoft OneLake Source

Source:
<https://docs.dremio.com/current/data-sources/object/s3/>

Version: current [26.x]

On this page

Amazon S3

This topic provides information for configuring the Amazon S3 data source.

Working with files stored in S3

You can query files and folders stored in your S3 buckets.

Dremio supports a number of different file formats.

See [Formatting Data to a Table](#) for more information.

note

Amazon S3 data sources added to projects default to using the Apache Parquet table format. Follow these steps to ensure that the default table format for new tables is Apache Iceberg:

In Dremio, click the Amazon S3 data source.

Click the gear icon in the top-right corner above the list of the data source's contents.

On the Advanced Options page of the Edit Source dialog, select **ICEBERG** under **Default CTAS Format**.

Click **Save**.

Amazon S3 data sources added to projects before Dremio 22 continue to use the Parquet table format for tables. For the SQL commands that you can use to create and query tables in such data sources, see [Tables](#).

Amazon Configuration

Amazon configuration involves:

Providing AWS credentials

Providing IAM Policy requirements

Amazon S3 Credentials

To list your AWS account's S3 buckets as a source, you must provide your AWS credentials in the form of your access and secret keys. You can find instructions for creating these keys in [Amazon's Access Key documentation](#).

AWS credentials are not necessary if you are accessing only public S3 buckets.

Sample IAM Policy for Accessing S3

The following sample IAM Policy shows the minimum policy requirements that allow Dremio to read and query S3.

Sample IAM policy for accessing Amazon S3

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1554422960000",
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListAllMyBuckets"
      ],
      "Resource": [
        "arn:aws:s3:::*"
      ]
    },
    {
      "Sid": "Stmt1554423012000",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::BUCKET-NAME"
      ]
    }
  ]
}
```

```

    },
    {
        "Sid": "Stmt1554423050000",
        "Effect": "Allow",
        "Action": [
            "s3:GetObject"
        ],
        "Resource": [
            "arn:aws:s3:::BUCKET-NAME/*"
        ]
    }
]
}

```

Sample IAM Policy for Writing to S3

The following sample IAM Policy shows the minimum policy requirements that allows Dremio to write to S3.

For example, to store Reflections on S3.

Sample IAM policy for writing to Amazon S3

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::BUCKET-NAME",
        "arn:aws:s3:::BUCKET-NAME/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets",
        "s3:GetBucketLocation"
      ],
      "Resource": "*"
    }
  ]
}

```

Configuring Amazon S3 as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Object Storage**, select **Amazon S3**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ``/`, `:`, `[`, or `]`.

Authentication

Choose one of the following authentication methods:

AWS Access Key: All or allowed (if specified) buckets associated with this access key or IAM role to assume, if provided, will be available. See Advanced Options for whitelisted information.

Under **AWS Access Key**, enter the [AWS access key ID](#).

Under **AWS Access Secret**, provide the [AWS access secret](#) using one of the following methods:

Dremio: Provide the access secret in plain text. Dremio stores the access secret.

Azure Key Vault: Provide the URI for your stored access set using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the access secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the access secret reference in the required format.

Under **IAM Role to Assume**, enter the [IAM role](#) that Dremio should assume in conjunction with the AWS Access Key authentication method.

EC2 Metadata: All or whitelisted (if specified) buckets associated with the IAM role attached to EC2 or IAM role to assume (if specified) will be available. See Advanced Options for whitelisted information.

Under **IAM Role to Assume**, enter the [IAM role](#) that Dremio should assume in conjunction with the EC2 Metadata authentication method.

EKS Pod Identity: Dremio can access all S3 buckets linked to the IAM role associated with the Kubernetes service account or the assumed IAM role. If you specify certain buckets, only those will be available.

Under **IAM Role to Assume**, enter the [IAM role](#) that Dremio should assume when using the Pod Identity authentication method.

AWS Profile: Dremio sources profile credentials from the specified AWS profile. For information on how to set up a configuration or credentials file for AWS, see [AWS Custom Authentication](#).

Profile Name (Optional): The AWS profile name. If this is left blank, then the default profile will be used. For more information about using profiles in a credentials or configuration file, see AWS's documentation on [Configuration and credential file settings](#).

No Authentication: Only the buckets provided under **Public Buckets** will be available.

The **Encrypt connection** option is enabled by default to encrypt the connection to S3. Clear the checkbox to disable encryption.

AWS Custom Authentication

[AWS Glue](#), [S3](#), and [Amazon OpenSearch](#) sources allow Dremio to use your AWS profile to authenticate users accessing your AWS-hosted data.

This authentication is performed by selecting the **AWS Profile** option for a source. Dremio will use credentials from the selected profile in the credentials file to authenticate with the source. Multiple methods are available for authentication, such as an external process. However, such processes must be created and validated for security by the users themselves.

note

We recommend using supported and secure methods via the AWS SDK and AWS application to minimize the potential for security risks.

Users with methods of generating or retrieving credentials that may not be supported by the AWS SDK can still use the tool by using additional configurations to alter the SDK, such as using the ``credential_process`` setting in the ``credentials`` file. Again, additional options are available for authenticating users via AWS. For more details regarding the storage of configuration settings and credentials maintained by AWS SDK, read AWS's [Configuration and credential file settings](#) documentation. This topic discusses both the supported settings available for inclusion in the configuration and credential files, as well as details regarding the storage of credentials.

Further information regarding this setting is found at AWS's documentation for [Sourcing credentials with an external process](#). This help topic outlines not only how to execute your command, but also how to structure the expected JSON-formatted output from a Credentials program, which Dremio requires.

Public Buckets

Add any external buckets that are not included with the provided AWS account credentials.

Advanced Options

Click **Advanced Options** in the left menu sidebar. All advanced options are optional.

Review each option provided in the following table to set up the advanced options to meet your needs.

Advanced Option Description

| --- | --- |

| **Enable compatibility mode** | Enables the use of S3-compatible storage such as MinIO. |

| **Apply requester-pays to S3 requests** | The requester (instead of the bucket owner) pays the cost of the S3 request and the data downloaded from the S3 bucket. |

| **Enable file status check** | Activated by default, uncheck the box to deactivate. Enables Dremio to check if a file exists in the S3 bucket before proceeding to handle errors gracefully. Disable this option when there are no files missing from the S3 bucket or when the file's access permissions have not changed. Disabling this option reduces the amount of communication to the S3 bucket. |

| **Enable partition column inference** | Enable this option to change how Dremio handles partition columns (see [Partition Column Inference](#) for more information.) |

| **Root Path** | The root path for the Amazon S3 bucket. The default root path is /. |

| **Server side encryption key ARN** | Add the ARN key created in [AWS Key Management Service](#) (KMS) if you want to store passwords in AWS KMS. Ensure that the AWS credentials that you share with Dremio have access to this ARN key. |

| **Default CTAS Format** | Choose the default format for tables you create in Dremio, either Parquet or Iceberg. |

| **Connection Properties** | Provide the custom key value pairs for the connection relevant to the source. 1. Click **Add Property**. 2. For Name, enter a connection property. 3. For Value, enter the corresponding connection property value. |

| **Allowlisted buckets** | Add an approved S3 bucket in the text field. You can add multiple S3 buckets this way. When using this option to add specific S3 buckets, you will only be able to see those buckets and not all the buckets that may be available in the source. Buckets entered must be valid. Misspelled or non-existent buckets will not appear in the resulting source. |

To configure your S3 source to use server-side encryption based on a provided key (SSE-C) or KMS (SSE-KMS), set the following connection properties:

SSE-C

``fs.s3a.server-side-encryption-algorithm`` set to ``SSE-C``

``fs.s3a.server-side-encryption.key`` set to the key used on the objects in S3

SSE-KMS

``fs.s3a.server-side-encryption-algorithm`` set to ``SSE-KMS``

``fs.s3a.server-side-encryption.key`` set to the ARN used on the objects in S3

Cache Options

Under Cache Options, review the following table and edit the options to meet your needs.

Cache Options	Description
---------------	-------------

| --- | --- |

Enable local caching when possible Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.	
---	--

Max percent of total available cache space to use when possible Specifies the disk quota, as a percentage, that a source can use on any single executor node only when local caching is enabled. The default is 100 percent of the total disk space	
--	--

available on the mount point provided for caching. You can either manually enter in a percentage in the value field or use the arrows to the far right to adjust the percentage.

Locations in which Iceberg Tables are Created

Where the CREATE TABLE command creates a table depends on the type of data source being used. For Amazon S3 sources, the root physical location is the main root directory for the filesystem. From this location, the path and table name are appended to determine the physical location for a new table.

caution

If your S3 datasets include large Parquet files with 100 or more columns, then you must edit the number of maximum connections to S3 that each processing unit of Dremio is allowed to spawn. To change the maximum connections:

Under Connection Properties, click **Add Property**.

For Name, enter ``fs.s3a.connection.maximum``.

For Value, enter a custom value greater than the default 100.

!Advanced Options !

Connecting through a proxy server

Optionally, you can configure your S3 source to connect through a proxy. You can achieve this by adding the following `Properties` in the settings for your S3 source:

Property Name	Description
---	---
fs.s3a.proxy.host	Proxy host.
fs.s3a.proxy.port	Proxy port number.
fs.s3a.proxy.username	Username for authenticated connections, optional.
fs.s3a.proxy.password	Password for authenticated connections, optional.

Connecting to a bucket in AWS GovCloud

To connect to a bucket in AWS GovCloud, set the correct GovCloud endpoint for your S3 source. You can achieve this by adding the following `Properties` in the settings:

Property Name	Description
---	---
fs.s3a.endpoint	The GovCloud endpoint (e.g., <code>`s3-us-gov-west-1.amazonaws.com`</code>).

Connecting to a bucket via AWS PrivateLink

To connect to a bucket using an AWS PrivateLink URL, set the correct server endpoint for your S3 source.

You can achieve this by adding the following `Properties` in the settings:

Property Name	Description
fs.s3a.endpoint.region	VPC region name (e.g., `us-east-1`).
fs.s3a.endpoint	PrivateLink DNS name (e.g., `bucket.vpce-xxx-xx.s3.us-east-1.vpce.amazonaws.com`).

note

The `fs.s3a.endpoint.region` setting ensures that the PrivateLink is created in the desired region, and it allows access only to buckets in the specified region.

The `fs.s3a.endpoint` value cannot contain the `http(s)://` prefix.

Reflection Refresh

 !

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

 !

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Automatically format files into tables when users issue queries.

If this box is checked and a query runs against the un-promoted table/folder,

Dremio automatically promotes using default options.

If you have CSV files, especially with non-default options, it might be useful to **not** check this box.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

note

Dataset Discovery is available for Google Cloud Storage (GCS) and databases. For Amazon S3, Dremio lists folders and files in real time unless they have been converted to tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Amazon S3 Source

To update an Amazon S3 source:

On the Datasets page, under **Object Storage** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see

Configuring Amazon S3 as a Source.

Click **Save**.

Deleting an Amazon S3 Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Amazon S3 source, perform these steps:

On the Datasets page, click **Sources > Object Storage** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Configuring S3-Compatible Storage

You can use S3-compatible storage, such as MinIO or IBM Cloud Object Storage, as a Dremio data source as long as the storage is completely S3-compatible. We recommend confirming S3 compatibility with the storage provider before you start the configuration steps.

To configure S3-compatible storage as a data source in the Dremio console:

Under **Advanced Options**, check **Enable compatibility mode**.

Under **Advanced Options > Connection Properties**, add ``fs.s3a.path.style.access`` and set the value to ``true``.

Note: This setting ensure that the request path is created correctly when using IP addresses or hostnames as the endpoint.

Under **Advanced Options > Connection Properties**, add the ``fs.s3a.endpoint`` property and its corresponding server endpoint value (IP address).

Limitation: The endpoint value cannot contain the ``http(s)://`` prefix nor can it start with the string ``s3``.

For example, if the endpoint is ``http://123.1.2.3:9000``, the value is ``123.1.2.3:9000``.

For IBM Cloud Object Storage and other S3-compatible storage where required, you must add the following property to the ``core-site.xml`` file to whitelist the bucket:

Whitelist the S3-compatible bucket

```
<property>
  <name>dremio.s3.whitelisted.buckets</name>
  <value>your-S3-compatible-bucket-name</value>
</property>
```

As an example for a specific S3-compatible storage product, the following steps describe how to configure your S3 source for MinIO with an encrypted connection in the Dremio console:

Use OpenSSL to generate a self signed certificate.

See [Securing Access to Minio Servers](#) or use an existing self signed certificate.

Start up Minio server with `./minio server [data folder] certs-dir [certs directory]`.

Install Dremio.

In your client environment where Dremio is located, install the certificate into **<JAVA_HOME>/jre/lib/security** with the following command:

Install certificate

```
<JAVA_HOME>/keytool -import -v -trustcacerts -alias alias -file cert-file -keystore
cacerts -keypass changeit -storepass changeit
```

note

Replace ``alias`` with the alias name you want and replace ``cert-file`` with the absolute path of the certificate file used to startup Minio server.

Start up Dremio.

In the Dremio console, add and configure an Amazon S3 data source with the Minio plug-in.

Under the **General** tab, specify the **AWS Access Key** and **AWS Access Secret** provided by your Minio server.

Under the **General** tab, check **Encrypt Connection**.

Under **Advanced Options**, check **Enable compatibility mode**.

Under **Advanced Options > Connection Properties**, add ``fs.s3a.path.style.access`` and set the value to ``true``.

Note: This setting ensure that the request path is created correctly when using IP addresses or hostnames as the endpoint.

Under **Advanced Options > Connection Properties**, add the ``fs.s3a.endpoint`` property and its corresponding server endpoint value (IP address).

Limitation: The endpoint value cannot contain the ``http(s)://`` prefix nor can it start with the string ``s3``.

For example, if the endpoint is ``http://123.1.2.3:9000``, the value is ``123.1.2.3:9000``.

Distributed Storage

Dremio requires object storage to be configured as [distributed storage](#). See the configuration of distributed storage for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information.

Configuring Minio as a Distributed Store

Minio can be used as a distributed store. Note that Minio works as a distributed store for both SSL and unencrypted connections. See the configuration of S3-compatible distributed storage for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information.

Configuring Cloud Cache

See [Configuring Cloud Cache](#) for more information.

Configuring KMS Encryption for Distributed Store

AWS Key Management Service (KMS) is available for S3 distributed store. See the configuration of distributed storage for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information.

For More Information

See the following Minio documentation for more information:

[Setting Up Minio](#)

[Loading Certificates in Minio](#)

[Securing Access to Minio Servers](#)

Was this page helpful?

[\[Previous](#)

[Object Storage\]\(/current/data-sources/object/\)\[Next](#)

[Azure Storage\]\(/current/data-sources/object/azure-storage\)](#)

[Working with files stored in S3](#)

[Amazon Configuration](#)

[Amazon S3 Credentials](#)

[Sample IAM Policy for Accessing S3](#)

[Sample IAM Policy for Writing to S3](#)

[Configuring Amazon S3 as a Source](#)

-

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an Amazon S3 Source

Deleting an Amazon S3 Source

Configuring S3-Compatible Storage

Distributed Storage

Configuring Minio as a Distributed Store

Configuring Cloud Cache

Configuring KMS Encryption for Distributed Store

For More Information

Source:
<https://docs.dremio.com/current/data-sources/object/azure-storage/>

Version: current [26.x]

On this page

Azure Storage

The Dremio Azure Storage Connector includes support for the following Azure Storage services:

Azure Blob Storage

Azure Blob storage is Microsoft's object storage solution for the cloud. Blob storage is optimized for storing massive amounts of unstructured data, such as text or binary data.

Azure Data Lake Storage Gen2

Azure Data Lake Storage Gen2 is a set of capabilities dedicated to big data analytics, built on top of Azure Blob storage. Features, such as file system semantics, directory, and file-level security and scale, are combined with the low-cost, tiered storage, and high availability/disaster recovery capabilities of Azure Blob storage.

note

Soft delete for blobs is not supported for Azure Storage accounts. Soft delete should be disabled to establish a successful connection.

Configuring Azure Storage as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Object Storage**, select **Azure Storage**.

General

Under **Name**, enter the name to use for the Azure Storage source. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Connection

Account Name: Name of the Azure Storage account.

Encrypt connection: Select to encrypt network traffic over SSL.

Storage Connection Protocol (Driver): Select the Azure Storage driver connection protocol you would like to use. The options are WASBS (Legacy) and ABFSS (Recommended). ABFSS is the default based on Azure best practices.

Authentication

Azure Storage authentication options include the following:

Shared access key: Select the secret store method from the dropdown menu:

Dremio: Provide the shared access key in plain text. Dremio stores the key.

Azure Key Vault: Provide the URI for your stored password using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the Azure Storage shared access key, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the shared access key secret reference in the required format.

Microsoft Entra ID:

Application ID: Specify the Application (Client) ID in Microsoft Entra ID.

OAuth 2.0 Token Endpoint: Specify the OAuth 2.0 token endpoint for your Azure application.

Application Secret Store: Select the secret store for the Application Secret from the dropdown menu:

Dremio: Provide the Application Secret in plain text. Dremio stores the key.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the Application Secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the application secret reference in the required format.

Advanced Options

Enable partition column inference: Select if Dremio should use partition column inference to handle partition columns.

Root Path: Root path for the source. The default is `/`.

Advanced Properties: Add connection properties, specifying their names and values.

Blob Containers & Filesystem Allowlist Add the names of containers to include in the source. This setting disables automatic container and filesystem discovery. Dremio limits the available containers and filesystems to those you add to the allowlist.

Cache Options

Enable local caching when possible: Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.

Max percent of total available cache space to use when possible: Maximum amount of cache space, as a percentage, that a source can use on any single executor node when local caching is enabled. The default value is `100`.

Reflection Refresh

The Reflection refresh options control how often Dremio refreshes Reflections automatically and the time limit after which Reflections expire and are removed.

Refresh Policy

Never refresh: Select to prevent the automatic refresh of Reflections. The default is to allow automatic refreshes.

Refresh every: If using automatic refresh, how often to refresh Reflections, specified in minutes, hours, days, or weeks. The default is 1 hour. Ignored if you select *Never refresh*.

Never expire: Select to prevent the expiration of Reflections. The default is expiration after the specified time limit.

Expire after: Time limit after which Reflections expire and are removed from Dremio, specified in minutes, hours, days, or weeks. The default is 3 hours. Ignored if you select *Never expire*.

Metadata

Metadata settings include options for dataset handling and metadata refresh.

Dataset Handling

Remove dataset definitions if underlying data is unavailable: Select to automatically remove datasets if their underlying files and folders are removed from Azure Storage or if the folder or source is not accessible. This option is selected by default. If **not** selected, Dremio does not remove dataset definitions even if their underlying files and folders are removed from Azure Storage, which is useful when files are temporarily deleted and replaced with a new set of files.

Automatically format files into tables when users issue queries: Select to automatically promote folders to tables using the default options when a user runs a query on the folder data for the first time. This option is not selected by default. For Azure Storage sources that contain CSV files, especially CSV files with non-default formatting, consider leaving this option unselected.

Metadata Refresh

Metadata Refresh settings allow you to configure the refresh interval for gathering detailed information about promoted tables, including fields, data types, shards, statistics, and locality. Dremio uses this information during query planning and optimization.

Fetch mode: The default is **Only Queried Datasets**, which only updates details only for previously queried objects in a source. This option increases query performance because the datasets require less work at query time. Other options are deprecated.

Fetch every: How often to refresh dataset details, specified in minutes, hours, days, or weeks. The default is 1 hour.

Expire after: Time limit after which dataset details expire, specified in minutes, hours, days, or weeks. The default is 3 hours.

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Azure Storage Source

To update an Azure Storage source:

-

On the Datasets page, under **Object Storage** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Azure Storage as a Source](#).

Click **Save**.

Deleting an Azure Storage Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Azure Storage source, perform these steps:

On the Datasets page, click **Sources > Object Storage** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Distributed Storage

Dremio requires object storage to be configured as [distributed storage](#). See the configuration of distributed storage for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#) for more information.

Azure Government

To configure Azure Storage for the Azure Government platform, add one of the following properties to the **Advanced Options** tab under **Advanced Properties**:

Storage V1: Add the following property and value if the Azure Storage source is of Account Kind Storage V1

Property and value for Storage V1

```
fs.azure.endpoint = blob.core.usgovcloudapi.net
```

Storage V2: Add the following property and value if the Azure Storage source is of Account Kind Storage V2

Property and value for Storage V2

```
fs.azure.endpoint = dfs.core.usgovcloudapi.net
```

Troubleshooting

If you see 0 byte files being created with your Iceberg tables in your Azure Storage account, these files do not impact Dremio's functionality and can be ignored if you cannot update your storage container. If you can update your container, see [Azure Data Lake Storage Gen2 hierarchical namespace](#) for more information on how to enable **Hierarchical Namespace** to prevent the creation of these files.

For More Information

[A closer look at Azure Data Lake Storage Gen2](#)

[Azure Data Lake Storage Gen2 Introduction](#)

[Azure Government cloud platform](#)

Was this page helpful?

[Previous

Amazon S3][(/current/data-sources/object/s3)[Next

Google Cloud Storage (GCS)][(/current/data-sources/object/gcs)

Configuring Azure Storage as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an Azure Storage Source

Deleting an Azure Storage Source

Distributed Storage

Azure Government

Troubleshooting

For More Information

Source:
<https://docs.dremio.com/current/data-sources/object/gcs/>

Version: current [26.x]

On this page

Google Cloud Storage (GCS)

Dremio allows for integration with environments using the Google Cloud Storage (GCS) web service for storing data. Configuration of this source allows for direct access to GCS data through the Dremio interface.

Configuring GCS as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Object Storage**, select **Google Cloud Storage**.

General

The following options are available from the *General* tab:

Name: Provide a name to identify the GCS data source in Dremio. The name cannot include the following special characters: `/`, `:`, `[`, or `.`.

Google Project Id: The specific ID for your GCS project. This can be found in the **Project info** pane at the top-left of your screen when at the Home page.

Authentication

Choose the authentication method:

Service Account Keys

Client Email: Provide the email address associated with the GCS service account.

Client ID: Provide the client ID for your key pair. The value is found by following the steps to create a service account key.

Private Key ID: Provide the key ID for your key pair. The value is found by following the steps to create a service account key.

Private Key: Choose a method for providing the private key for your key pair. The value is found by following the steps to create a service account key.

Dremio: Provide the private key in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored private key using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets

Manager secret that holds the private key, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the private key secret reference in the required format.

Automatic/Service Account: If you are currently running Dremio on a Google Compute instance, Dremio uses the active service account for your GCS data source and does not require any additional information to integrate with your data.

Creating Service Account Keys

In order to use Dremio to access your Google Cloud Storage source, you need to first identify the service account. This is done by creating public/private key pairs. When creating service account keys, the public portion is stored on Google Cloud, while the private portion is made available to you for entry on Dremio.

The steps below outline the most simple method of creating a service account key.

From the Google Cloud Console, navigate to the Service Accounts page.

Select the desired project.

Click on the email address of the service account that you'll be creating a key for.

Click on the ***Keys*** tab.

Click the **Add Key** drop-down menu and then select **Create new key**.

Select **JSON** as the **Key Type** and then click **Create**.

Your browser then downloads a service account key file. It should look similar to the example below:

Example service account key file

```
{
  "type": "service_account",
  "project_id": "project-id",
  "private_key_id": "key-id",
  "private_key": "-----BEGIN PRIVATE KEY-----\nprivate-key\n-----END PRIVATE KEY-----\n",
  "client_email": "service-account-email",
  "client_id": "client-id",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://accounts.google.com/o/oauth2/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/service-account-email"
}
```

Copy and paste each value from this file to the corresponding fields on the Dremio interface.

For additional methods of creating a key (e.g., ``gcloud`` tool, REST APIs, etc.), [view](#)

Advanced Options

The following settings control more advanced functionalities in Dremio.

Field	Description
---	---
Root Path	The root path for the GCS source.
Properties	Additional connection properties, consisting of the property and its specified value.
Whitelisted buckets	A list of buckets to whitelist, or allow access to.
Cache Options	
Enable local caching when possible	Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.
Max percent of total available cache space to use when possible	Sets the allowable amount of local caching, based on percentage. Only the percent specified of the cached files will be stored locally. By default, this is set to 100.

Reflection Refresh

This tab controls the frequency of Reflection refreshes or the timespan for expiration for any queries performed using this data source.

Field	Description
---	---
Never refresh	Prevents any query Reflections associated with this source from refreshing.
Refresh every	Sets the time interval by which Reflections for this source are refreshed. This may be set to hours, days, and weeks.
Never expire	Prevents any query Reflections associated with this source from expiring.
Expire after	Sets the time after a Reflection is created that it then expires and can no longer be used for queries. This may be set to hours, days, and weeks.

Metadata

This tab offers settings that control how dataset details are fetched and refreshed.

Field	Description
---	---
Dataset Handling	
Remove dataset definitions if underlying data is unavailable	If this box is not checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.
Automatically format files into tables when users issue queries	If this box is checked and a query runs against the un-promoted table/folder, Dremio automatically promotes using default options. If you have CSV files, especially with non-default options, it might be useful to not check this box.
Metadata Refresh	

| Dataset Discovery | Specifies the refresh interval for top-level source object names, such as database and table names. This is a lightweight operation. Fetch every. Specifies the time interval by which Dremio fetches object names. This can be set by minutes, hours, days, and weeks. |

| Dataset Details | Specifies the metadata that Dremio needs for query planning, such as information regarding fields, types, shards, statistics, and locality. Fetch mode. Restricts when metadata is retrieved. Only Queried Datasets. Dremio updates metadata details for previously-queried objects in a source. This mode increases query performance as it requires less work to be done at query time for these datasets. All Datasets (deprecated). Dremio updates the details for all datasets in a source. This mode increases query performance as less work is needed to be done at the time of query. Fetch every. Specifies the time interval by which metadata is fetched. This can be set by minutes, hours, days, and weeks. Expire after. Specifies the timespan for when dataset details expire after a dataset is queried. This can be set by minutes, hours, days, and weeks. |

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a GCS Source

To update a GCS source:

On the Datasets page, under **Object Storage** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring GCS as a Source](#).

Click **Save**.

Deleting a GCS Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can

delete the source.

To delete a GCS source, perform these steps:

On the Datasets page, click **Sources** > **Object Storage** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[Previous

Azure Storage](/current/data-sources/object/azure-storage)[Next

HDFS](/current/data-sources/object/hdfs)

Configuring GCS as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a GCS Source

Deleting a GCS Source

Source:
<https://docs.dremio.com/current/data-sources/object/hdfs/>

Version: current [26.x]

On this page

HDFS

This topic describes HDFS data source considerations and Dremio configuration.

HBase

HBase is an open-source, non-relational database that is built on top of HDFS and enables real-time analysis of data.

note

Although HBase is no longer officially supported by Dremio as a source connection, you can still add HBase as a Dremio source by using a [community connector](#).

Files stored in HDFS

You can query files and folders stored in your HDFS cluster. Dremio supports a number of different file formats.

See [Formatting Data to a Table](#) for more information.

Co-location

Co-locating Dremio nodes with HDFS datanodes can lead to noticeably reduced data transfer times and more performant query execution.

Parquet File Performance

When HDFS data is stored in the Parquet file format, then optimal performance is achieved

by storing one Parquet row group per file, with a file size less than or equal to the HDFS block size.

Parquet files that overrun the HDFS block size can negatively impact query times by incurring a considerable amount of filesystem overhead.

note

Ensure that your Dremio cluster has access to the appropriate ports for each node of your HDFS source. By default, this should be port 8020 for an HDFS NameNode (which should be the one specified when adding the source), and either port 50010 or port 9866 for HDFS DataNodes (dfs.datanode.address, used internally for data transfer).

HDFS Configuration

This section provides HDFS configuration.

Impersonation

To grant the Dremio service user the privilege to connect from any host and to impersonate a user belonging to any group, modify the **core-site.xml** file with the following values:

User impersonation settings for core-site.xml file

```
<property>
  <name>hadoop.proxyuser.dremio.hosts</name>
  <value>*</value>
</property>
```

```
<property>
  <name>hadoop.proxyuser.dremio.groups</name>
  <value>*</value>
</property>
<property>
  <name>hadoop.proxyuser.dremio.users</name>
  <value>*</value>
</property>
```

To modify the properties to be more restrictive by passing actual hostnames and group names, modify the **core-site.xml** file with the following values:

More restrictive user impersonation settings for core-site.xml file

```
<property>
  <name>hadoop.proxyuser.super.hosts</name>
  <value>10.222.0.0/16,10.113.221.221</value>
</property>
<property>
  <name>hadoop.proxyuser.dremio.users</name>
  <value>user1,user2</value>
</property>
```

Impersonation and Privilege Delegation

You can enable user-specific file access permissions by turning on impersonation in HDFS sources

(check the 'impersonation' box in the source connection dialog).

Users who access data stored on an HDFS source with impersonation enabled will have their access mediated by the

HDFS privileges associated with their Dremio login name, rather than the ones associated with the Dremio daemon.

For example, let's say a Dremio user named `bobsmith` has been granted read access to the

file `/accounts/CustomerA.txt` under the same username in HDFS. However, the `dremio` system user

(the user that the Dremio daemon runs as) does not have read access to this file.

Unless impersonation was enabled when this HDFS source was added to Dremio, `bobsmith` will be unable to query the file.

Enabling impersonation also permits a kind of behavior called 'privilege delegation.'

Under privilege delegation, HDFS data which is subject to restricted access can be shared

with any other Dremio users via the creation of a view in a public (non-Home) space.

NameNode HA Configuration

If you have configured a secondary NameNode and a Dremio HA configuration, you must configure Dremio to reconnect with the secondary NameNode in the event

the first NameNode goes down.

To configure a secondary NameNode:

Ensure that `fs.defaultFs` parameter and value is specified in the **core-site.xml** file *without* the port number.

(The port is already specified in the URI.)

Specify `fs.defaultFs` parameter and value

```
<name>fs.defaultFS</name>
<value>hdfs://xyzcluster</value>
```

Configure the NameNode HA parameters via one of the following methods:

Copy/symlink the Hadoop **core-site.xml** file to the Dremio **conf** folder if you haven't already done so.

Add the following parameters and values to the HDFS source in the Dremio UI under Advanced Options.

HDFS source parameters and values

```
dfs.nameservices - (say this value is my cluster)
dfs.ha.namenodes.mycluster - (say this value is nn1, nn2)
dfs.namenode.rpc-address.mycluster.nn1
dfs.namenode.rpc-address.mycluster.nn2
dfs.client.failover.proxy.provider.mycluster
```

(Optional) Configure your distributed storage to **hdfs** in the `dremio.conf` file.

For more information on NameNode HA in Cloudera or Hortonworks, see:

[Setting up NameNode HA on Cloudera](#)

[Setting up NameNode HA on Hortonworks](#)

Configuring HDFS as a Source

The HDFS source is usually configured when you are adding a new source, especially the name and connection parameters. However, additional options can be changed or added by editing an existing source.

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Object Storage**, select **HDFS**.

General

Name HDFS Name for the source. The name cannot include the following special characters: `\`, `:`, `[`, or `]`.

-

Connection HDFS connection and impersonation

NameNode Host

No HA - HDFS NameNode hostname.

HA - value for `dfs.nameservices` from `hdfs-site.xml`.

NameNode Port HDFS NameNode port

Enable Impersonation When enabled, Dremio executes queries against HDFS on behalf of the user.

When **Allow VDS-based Access Delegation** is enabled (default), the owner of the view is used as the impersonated username.

When **Allow VDS-based Access Delegation** is disabled (unchecked), the query user is used as the impersonated username.

Advanced Options

The advanced options tab has the following values:

Enable exports into the source (CTAS and DROP)

Root Path Root path for the HDFS source

Short-Circuit Local Reads Implementation of short-circuit local reads on which clients directly open the HDFS block files.

HDFS Default

Enabled

Disabled (Default)

Impersonation User Delegate Specifies whether an impersonation username is one of the following:

As is (Default)

Lowercase

Uppercase

Connection Properties A list of additional HDFS connection properties.

Cache Options

Enable local caching when possible Selected by default, along with asynchronous access for cloud caching, local caching can improve query performance. See Cloud Columnar Cache for details.

Max percent of total available cache space to use when possible. Default: 100

Reflection Refresh

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Automatically format files into tables when users issue queries.

If this box is checked and a query runs against the un-promoted table/folder,

Dremio automatically promotes using default options.

If you have CSV files, especially with non-default options, it might be useful to **not** check this box.

Metadata Refresh

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Authorization When impersonation is enabled, the maximum amount of time that Dremio will cache authorization information.

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an HDFS Source

To update an HDFS source:

On the Datasets page, under **Object Storage** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring HDFS as a Source](#).

Click **Save**.

Deleting an HDFS Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an HDFS source, perform these steps:

On the Datasets page, click **Sources > Object Storage** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[Previous

Google Cloud Storage (GCS)](/current/data-sources/object/gcs)[Next

NAS](/current/data-sources/object/nas)

HBase

Files stored in HDFS

Co-location

Parquet File Performance

HDFS Configuration

Impersonation

Impersonation and Privilege Delegation

NameNode HA Configuration

Configuring HDFS as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an HDFS Source

Deleting an HDFS Source

Source:
<https://docs.dremio.com/current/data-sources/object/nas/>

Version: current [26.x]

On this page

NAS

Working with files and folders in your NAS

If your Dremio cluster is connected to your NAS, you can query folders and files stored in this data source.

All nodes in your Dremio cluster should be able to connect to your NAS.

Configuring NAS as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Object Storage**, select **NAS**.

General

Name Enter the name to identify the data source in Dremio. The name cannot include the following special characters: ```, `:`, `['`, or `']`.

Mount Path Path on the filesystem to use as the root for the source. Needs to be accessible on all nodes.

Advanced Options

 !

Enable exports into the source (CTAS and DROP).

Reflection Refresh

![/assets/images/hdfs-refresh-policy-9ae71114907887b859a9d01425390739.png) !

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

![/assets/images/nas-metadataD-404041022ad0b766a1ab8584df7b0668.png) !

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Automatically format files into tables when users issue queries.

If this box is checked and a query runs against the un-promoted table/folder,

Dremio automatically promotes using default options.

If you have CSV files, especially with non-default options, it might be useful to **not** check this box.

Metadata Refresh

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an NAS Source

To update an NAS source:

On the Datasets page, under **Object Storage** in the panel on the left, find the name of the source you want to edit.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring NAS as a Source](#).

Click **Save**.

Deleting an NAS Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an NAS source, perform these steps:

On the Datasets page, click **Sources > Object Storage** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[Previous

HDFS](/current/data-sources/object/hdfs)

Working with files and folders in your NAS

Configuring NAS as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an NAS Source

Deleting an NAS Source

Source:
<https://docs.dremio.com/current/data-sources/databases/opensearch>

Version: current [26.x]

On this page

Amazon OpenSearch Service

Amazon OpenSearch Service is a managed service that makes it easy to deploy, operate, and scale OpenSearch clusters in the AWS Cloud.

Compatibility

Dremio supports the following Amazon OpenSearch Service versions:

5.x

6.0

6.2

6.3

7.0+

Amazon OpenSearch is supported as a data source in Dremio Software on-premises deployments.

Configuring Amazon OpenSearch Service as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Amazon OpenSearch Service**.

General

On the General tab, enter a name for the source, connection details, and authentication credentials. The name cannot include the following special characters: `\`, \:`, \[`, or \]``.

Connection

Name	Description
Host	AWS OpenSearch Host name.
Port	Port on which the AWS OpenSearch service is running (usually 443).

Authentication

Choose one of the following authentication methods:

AWS Access Key: Used for key-based authentication.

Under **AWS Access Key**, enter the AWS access key ID.

Under **AWS Access Secret**, store the AWS access secret using one of the following methods:

Dremio: Provide the secret in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the secret reference in the required format.

EC2 Metadata: Dremio uses the IAM policy from the EC2 instance.

EKS Pod Identity: Dremio uses the IAM policy associated with the coordinator's Kubernetes service account.

AWS Profile: Dremio sources profile credentials from the specified AWS profile. For information on how to set up a configuration or credentials file for AWS, see AWS Custom Authentication.

Under AWS Profile (Optional), enter the AWS profile name. If this is left blank, then the default profile will be used. For more information about using profiles in a credentials or configuration file, see AWS's documentation on Configuration and credential file settings.

No Authentication: No credentials required.

Select the option to perform keyword searches when pushing down fields mapped as text and keyword if desired.

Advanced Options

On the Advanced Options tab, enter the options specific to the OpenSearch Service, encryption, and AWS.

OpenSearch options

Show hidden indices that start with a dot (.).

Use Painless scripting with OpenSearch 5.0+ (Checked as a default).

Show `_id` columns.

Use index/doc fields when pushing down aggregates and filters on analyzed and normalized fields (may produce unexpected results).

Use scripts for query pushdown`**` (Checked as a default).

If the number of records returned from OpenSearch is less than the expected number, warn instead of failing the query.

Read timeout (seconds) (default: 60)

Scroll timeout (seconds) (default: 300)

Scroll size This setting must be less than or equal to your OpenSearch value for the `_index.max_result_window`` setting. (default: 4000)

Encryption

Validation modes include:

Validate certificate and hostname (default)

Validate certificate only

Do not validate certificate or hostname

AWS

Overwrite region If the box is checked, provide the region.

Reflection Refresh

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.
This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.
This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.
This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Amazon OpenSearch Service Source

To update an Amazon OpenSearch Service source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively,

click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Amazon OpenSearch Service as a Source](#).

Click **Save**.

Deleting an Amazon OpenSearch Service Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Amazon OpenSearch Service source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Was this page helpful?

[\[Previous](#)

[Databases\]\(/current/data-sources/databases/\)](#)[\[Next](#)

[Amazon Redshift\]\(/current/data-sources/databases/redshift\)](#)

[Compatibility](#)

[Configuring Amazon OpenSearch Service as a Source](#)

[General](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Metadata](#)

[Privileges](#)

[Updating an Amazon OpenSearch Service Source](#)

Source:
<https://docs.dremio.com/current/data-sources/databases/redshift>

Version: current [26.x]

On this page

Amazon Redshift

Configuring Amazon Redshift as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Amazon Redshift**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ```, `:`, `[`, or `]`.

Connection

JDBC Connection String [Connection string. The connection URL can be found in AWS console.](#)

Authentication

Select an authentication option:

No Authentication

Master Credentials (default):

Username: Redshift username

Password: Select the password store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

[Azure Key Vault](#): Provide the URI for your stored password using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

[AWS Secrets Manager](#): Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

[HashiCorp Vault](#): Select your HashiCorp secrets engine from the dropdown and enter

the password reference in the correct format.

Secret Resource URL: Provide the Username and Secret Resource URL to use for authentication.

EKS Pod Identity: Dremio uses the IAM policy associated with the coordinator's Kubernetes service account.

AWS Profile: Dremio sources profile credentials from the specified AWS profile. For information on how to set up a configuration or credentials file for AWS, see [AWS Custom Authentication](#).

AWS Profile (Optional): The AWS profile name. If this is left blank, then the default profile will be used. For more information about using profiles in a credentials or configuration file, see AWS's documentation on [Configuration and credential file settings](#).

DbUser (Optional): The name of the Redshift DbUser to use for authentication. If this is left blank, the default user name for your AWS IAM role will be used (generally this is the same as your AWS username).

Advanced Options

 !

Record fetch size: Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. Default: 10

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to *8*.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to *60*.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Enable legacy dialect

Reflection Refresh

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Dataset Handling

-

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Amazon Redshift Source

To update an Amazon Redshift source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Amazon Redshift as a Source](#).

Click **Save**.

Deleting an Amazon Redshift Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Amazon Redshift source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

```
`*`,`+`,`-`,`/`  
`<`,`<=`,`<>`,`=`,`>`,`>=`,`!=`  
AND, NOT, OR, `||`  
ABS  
ACOS  
ADD_MONTHS  
ASIN
```

ATAN
ATAN2
AVG
CAST
CBRT
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COT
DATE_ADD
DATE_SUB
DATE_TRUNC_CENTURY
DATE_TRUNC_DAY
DATE_TRUNC_DECADE
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER
DATE_TRUNC_SECOND
DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DEGREES
E
EXP
EXTRACT_CENTURY
EXTRACT_DAY
EXTRACT_DECADE
EXTRACT_DOW
EXTRACT_DOY
EXTRACT_EPOCH
EXTRACT_HOUR
EXTRACT_MILLENNIUM
EXTRACT_MINUTE
EXTRACT_MONTH
EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE
LOG

LOG10
LOWER
LPAD
LTRIM
MAX
MEDIAN
MIN
MOD
PERCENT\ _CONT
PERCENT\ _DISC
PI
POSITION
POW
POWER
RADIANS
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
STDDEV
STDDEV\ _POP
STDDEV\ _SAMP
SUBSTR
SUBSTRING
SUM
TAN
TIMESTAMPADD\ _DAY
TIMESTAMPADD\ _HOUR
TIMESTAMPADD\ _MINUTE
TIMESTAMPADD\ _MONTH
TIMESTAMPADD\ _QUARTER
TIMESTAMPADD\ _SECOND
TIMESTAMPADD\ _WEEK
TIMESTAMPADD\ _YEAR
TIMESTAMPDIFF\ _DAY
TIMESTAMPDIFF\ _HOUR
TIMESTAMPDIFF\ _MINUTE
TIMESTAMPDIFF\ _MONTH
TIMESTAMPDIFF\ _QUARTER
TIMESTAMPDIFF\ _SECOND
TIMESTAMPDIFF\ _WEEK
TIMESTAMPDIFF\ _YEAR
TO\ _CHAR
TO\ _DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER

VAR_POP
VAR_SAMP

Running Queries Directly on Redshift Through Dremio

Dremio users can run pass queries through Dremio to run on Redshift. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

For More Information

See [Redshift Data Types](#)
for information about mapping to Dremio data types.

Was this page helpful?

[Previous

Amazon OpenSearch Service](/current/data-sources/databases/opensearch)[Next

Apache Druid](/current/data-sources/databases/apache-druid)

Configuring Amazon Redshift as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an Amazon Redshift Source

Deleting an Amazon Redshift Source

Predicate Pushdowns

Running Queries Directly on Redshift Through Dremio

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/apache-druid>

Version: current [26.x]

On this page

Apache Druid

Apache Druid is a high performance, real-time analytics database that delivers sub-second queries on streaming and batch data at scale and under load.

Prerequisite

Ensure that your Dremio cluster is at version 24.2 or later.

Configuring Apache Druid as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Druid**.

General

In the **Name** field, specify the name by which you want the Druid source to appear in the list of data sources. The name cannot include the following special characters: ``/``, ``:``, ``[``, or ``]``.

Under **Connection**, follow these steps:

In the **Host** field, specify the hostname or IP address of the Druid source.

In the **Port** field, specify the port to use. The default port is 8888.

(Optional) Select **Use SSL** to use SSL to secure connections.

Under **Authentication**, specify the Apache Druid username. Then, choose a method for storing the Apache Druid password from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown menu and enter the secret reference in the required format.

Advanced Options

On the Advanced Options page, you can set values for these non-required options:

Option	Description
---	---
Maximum Idle Connections	The total number of connections allowed to be idle at a given time. The default maximum idle connections is 8.
Connection Idle Time	The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. The default connection idle time is 60

seconds. |

| **Query Timeout** | The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state. |

Reflection Refresh

On the Reflection Refresh page, set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed.

| Option | Description |

| --- | --- |

| **Never refresh** | Select to prevent automatic Reflection refresh, default is to automatically refresh. |

| **Refresh every** | How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if **Never refresh** is selected. |

| **Never expire** | Select to prevent Reflections from expiring, default is to automatically expire after the time limit below. |

| **Expire after** | The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if **Never expire** is selected. |

Metadata

On the Metadata page, you can configure settings to refresh metadata and handle datasets.

Dataset Handling

These are the optional **Dataset Handling** parameters.

| Parameter | Description |

| --- | --- |

| **Remove dataset definitions if underlying data is unavailable** | By default, Dremio removes dataset definitions if underlying data is unavailable. Useful when files are temporarily deleted and added back in the same location with new sets of files. |

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

| Parameter | Description |

| --- | --- |

| (Optional) **Fetch every** | You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour. |

Dataset Details: The metadata that Dremio needs for query planning such as information required for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

| Parameter | Description |

| --- | --- |

| **Fetch mode** | You can choose to fetch only from queried datasets that are set by default. Dremio updates details for previously queried objects in a source. Fetching from all datasets is deprecated. |

| **Fetch every** | You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour. |

| **Expire after** | You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours. |

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Druid Source

To update a Druid source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Apache Druid as a Source](#).

Click **Save**.

Deleting a Druid Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Druid source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

These operations are performed by Druid:

`!=`
`*`
`+`
`-`
`/`
`<`
`<=`
`<>`
`= `
`>`
`>=`
abs
acos
and
asin
atan
atan2
avg
cast
ceil
concat
convert_timezone
cos
cot
degrees
floor
is not null
is null
length
like
ln
log
lower
lpad
ltrim
max
min

mod
not
or
power
radians
regexp_like
replace
reverse
round
rpad
rtrim
sign
sin
substr
substring
sum
tan
tanh
trim
upper
`||`

Was this page helpful?

[Previous

Amazon Redshift](/current/data-sources/databases/redshift)[Next

Dremio Cluster](/current/data-sources/databases/dremio)

Prerequisite

Configuring Apache Druid as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Druid Source

Deleting a Druid Source

Predicate Pushdowns

Source:
<https://docs.dremio.com/current/data-sources/databases/dremio>

Version: current [26.x]

Connecting to Another Dremio Software Cluster

You can add a Dremio Software cluster as a data source. Such a cluster is referred to as a **data-source cluster**. The Dremio cluster that you add it to is referred to as a **querying cluster**.

caution

Only Dremio Software can serve as a data-source cluster. Using Dremio Cloud as a data-source cluster is not supported.

A data-source cluster gives a querying cluster access to one or more data sources, such as Amazon S3, Hive, and Postgres, that are connected to the data-source cluster. Dremio treats the connected Dremio cluster as any other supported data source. The data sources that are connected to the data-source cluster are represented as schemas. From a querying cluster, you can drill down into the schemas to see source datasets. You can then promote source datasets to tables, create Reflections and views on those tables, and views on the views, and so on.

Example

In this diagram, there are two Dremio Software clusters: **Dremio_1** and **Dremio_2**. Suppose that you wanted to access **DataSource_2** from **Dremio_1**. To do so, you would add **Dremio_2** as a data source to **Dremio_1**. In fact, you could add any number of Dremio Software clusters as data sources.

!Connecting one Dremio Software cluster to another

In the UI for **Dremio_1**, **Dremio_2** is listed under **Sources > Databases**. If you were to select **Dremio_2** there, you would see the folder **DataSource_2**. Double-clicking that folder would show a list of the folders or schemas in that data source.

An administrator can promote a table on a data source connected through a data-source cluster, just as it is possible to do on data source that is directly connected to a querying cluster. For example, an administrator promotes table ``DataSource_1.Table_1`` from ``DataSource_1.Source_1`` on the data source directly connected to **Dremio_1**, and also promotes table ``Dremio_2.DataSource_2.Table_2`` from ``DataSource_2.Source_2`` via the data-source cluster.

!Connecting to a data source directly and through a Dremio Software cluster

The administrator can then use the tables as any other table, by querying them, creating views on them, and creating Reflections on them.

If **Dremio_1** were connected to two Dremio clusters, the administrator could promote tables on both. Then, business users could run queries and view reports that federated data across the two data-source clusters.

!Queries can federate data that is in two or more clusters.

And while business users run queries through the querying-cluster, other business users can continue running queries directly through a data-source cluster.

!Business users can still query data-source clusters directly.

caution

Ensure that the Dremio instance that you connect to does not itself connect to your original Dremio instance. For example, if **Dremio_1** connects to **Dremio_2** as a data source, ensure that **Dremio_2** does not connect to **Dremio_1** as a data source.

!Avoid recursive loops.

Querying across more than one region or more than one cloud vendor might increase query latency. Querying across cloud vendors also might result in egress charges from cloud vendors. For example, in this diagram **DataSource_1** is using one cloud vendor, while **DataSource_2** is using a different cloud vendor. Queries from **Dremio_1** across **Dremio_2** and **Dremio_3** against those two data sources might incur egress charges from the cloud vendors.

!Federating data across cloud vendors can incur egress charges.

tip

Dremio recommends full TLS wire encryption on querying clusters and data-source clusters. For more information, see the configuration of TLS for [Dremio on Kubernetes](#) or [Dremio standalone clusters](#).

User Impersonation

When you connect a querying cluster to a data-source cluster, you provide the username and password of an account on the data-source cluster. By default, queries that run from the querying cluster against the data-source cluster run under the username of that account.

You can instead allow users running queries from the querying cluster to run them under their own usernames on the data-source cluster. For example, User 1 on the querying cluster Dremio 1 can run queries as User 1 on the data-source cluster. User 1 must have an account on the data-source cluster, and that account must use the same username. User impersonation (also known as *inbound impersonation*) must be set up on the data-source cluster. The policy for user impersonation would look like this:

Example policy

```
ALTER SYSTEM SET "exec.impersonation.inbound_policies"='[
  {
    "proxy_principals":{
      "users":[
        "User 1"
      ]
    },
    "target_principals":{
      "users":[
        "User 1"
      ]
    }
  }
]
```

See [Inbound Impersonation](#) for more information.

Limitation

You cannot query columns that use complex data types, such as LIST, STRUCT, and MAP. Columns of complex data types do not appear in result sets.

Prerequisites

You must have a username and password for the account on a data-source cluster to use for connections from the querying cluster.

The querying cluster and data-source clusters must all be at version 23.1 or later.

Configuring Another Dremio Software Cluster as a Source

note

If the cluster that you are connecting to has a self-signed certificate, ensure that the cluster that you are connecting from has a copy of that certificate in its truststore.

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Dremio**.

General Options

In the **Name** field, specify the name by which you want the data-source cluster to appear in the **Databases** section. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Under **Connection**, specify how you want to connect to the data-source cluster.

Direct: Connect directly to a coordinator node of the cluster.

ZooKeeper: Connect to an external ZooKeeper instance that is coordinating the nodes of the cluster.

In the **Host** and Port **field**, specify the hostname or IP address, and the port number, of the coordinator node or ZooKeeper instance.

If the data-source cluster is configured to use TLS for connections to it, select the **Use SSL** option.

Under **Authentication**, specify the username for the querying cluster to use when connecting to the data-source cluster. Then, choose a method for storing the password for the querying cluster from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format

`https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the secret reference in the required format.

Advanced Options

On the Advanced Options page, you can set values for these non-required options:

- | Option | Description |
- | --- | --- |
- | **Maximum Idle Connections** | The total number of connections allowed to be idle at a given time. The default maximum idle connections is 8. |
- | **Connection Idle Time** | The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. The default connection idle time is 60 seconds. |
- | **Query Timeout** | The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state. |
- | **User Impersonation** | Allows users to run queries on the data-source cluster under their own user IDs, not the user ID for the account used to authenticate with the data-source cluster. Inbound impersonation must be configured on the data-source cluster. See Inbound Impersonation. |

Reflection Refresh Options

On the Reflection Refresh page, set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed.

- | Option | Description |
- | --- | --- |
- | **Never refresh** | Select to prevent automatic Reflection refresh, default is to automatically refresh. |
- | **Refresh every** | How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if **Never refresh** is selected. |
- | **Never expire** | Select to prevent Reflections from expiring, default is to automatically expire after the time limit below. |
- | **Expire after** | The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if **Never expire** is selected. |

Metadata Options

On the Metadata page, you can configure settings to refresh metadata and handle datasets.

Dataset Handling

These are the optional **Dataset Handling** parameters.

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

---	---
-----	-----

Remove dataset definitions if underlying data is unavailable	By default, Dremio removes dataset definitions if underlying data is unavailable. Useful when files are temporarily deleted and added back in the same location with new sets of files.
---	---

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

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

---	---
-----	-----

(Optional) Fetch every	You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.
-------------------------------	---

Dataset Details: The metadata that Dremio needs for query planning such as information required for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

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

---	---
-----	-----

Fetch mode	You can choose to fetch only from queried datasets that are set by default. Dremio updates details for previously queried objects in a source. Fetching from all datasets is deprecated.
-------------------	--

Fetch every	You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.
--------------------	---

Expire after	You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.
---------------------	--

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges.

note

All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Dremio Source

To update a Dremio source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Another Dremio Software Cluster as a Source](#).

Click **Save**.

Deleting a Dremio Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Dremio source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Querying clusters offload these operations to data-source clusters. Data-source clusters either process these operations or offload them to their connected data sources.

```
`&&`, `||`, `!`, `AND`, `OR`  
`+`, `-`, `/`, `*`, `%`  
`<=`, `<`, `>`, `>=`, `=`, `<>`, `!=`  
ABS  
ADD\_MONTHS  
AVG  
BETWEEN  
CASE  
CAST  
CEIL
```


CEILING
CHARACTER_LENGTH
CHAR_LENGTH
COALESCE
CONCAT
CONTAINS
COUNT
COUNT_DISTINCT
COUNT_DISTINCT_MULTI
COUNT_FUNCTIONS
COUNT_MULTI
COUNT_STAR
CURRENT_DATE
CURRENT_TIMESTAMP
DATE_ADD
DATE_DIFF
DATE_SUB
DATE_TRUNC
DATE_TRUNC_DAY
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER
DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DAYOFMONTH
DAYOFWEEK
DAYOFYEAR
EXTRACT
FLATTEN
FLOOR
ILIKE
IN
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE
LOCATE
LOWER
LPAD
LTRIM
MAX
MEDIAN
MIN
MOD
NEXT_DAY
NOT
NVL
PERCENTILE_CONT

PERCENTILE\ _DISC
PERCENT\ _RANK
POSITION
REGEXP\ _LIKE
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SQRT
STDDEV
STDDEV\ _POP
STDDEV\ _SAMP
SUBSTR
SUBSTRING
SUM
TO\ _CHAR
TO\ _DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
VAR\ _POP
VAR\ _SAMP

Related Information

[ZooKeeper](#)

Was this page helpful?

[\[Previous](#)

[Apache Druid\]\(/current/data-sources/databases/apache-druid\)\[Next](#)

[Elasticsearch\]\(/current/data-sources/databases/elasticsearch\)](#)

[Example](#)

[User Impersonation](#)

[Limitation](#)

[Prerequisites](#)

[Configuring Another Dremio Software Cluster as a Source](#)

[General Options](#)

[Advanced Options](#)

[Reflection Refresh Options](#)

Metadata Options

Privileges

Updating a Dremio Source

Deleting a Dremio Source

Predicate Pushdowns

Related Information

Source:
<https://docs.dremio.com/current/data-sources/databases/google-bigquery>

Version: current [26.x]

On this page

Google BigQuery Enterprise

Dremio supports connecting to Google BigQuery as an external source. The connector uses Google service account keys for authentication. To learn more about creating and managing service account keys, see [Create and delete service account keys](#).

Requirements

To connect to Google BigQuery, you need:

Google BigQuery

Source configuration for authentication

User Impersonation

Dremio supports authentication using Google Workforce Identity Pool impersonation, which allows external identities to securely access BigQuery datasets without requiring a dedicated Google service account for each user.

note

Reflections are not supported on data sources with user impersonation enabled to ensure that all security and governance policies defined in the underlying data source are enforced.

Reflections created prior to enabling user impersonation must be manually dropped, as they will fail to refresh once impersonation is active.

Prerequisites

Before configuring a Bigquery source with user impersonation, ensure you have:

Access to a Google Cloud Organization

An Organization Admin role (`roles/iam.organizationAdmin`) or Workforce Pool Admin role (`roles/iam.workforcePoolAdmin`) within that organization

Configure Google Workforce Identity Pool

In the following steps, you will configure Google Cloud to recognize and verify assertions (signed JWTs) sent from Dremio. Google uses Dremio's public key to validate the digital signatures of these assertions. Creating a Workforce Identity Pool establishes trust between Google Cloud and Dremio for OAuth-based authentication.

To allow federated identities from your Workforce Identity Pool to execute BigQuery jobs, you must assign the BigQuery Job User role (`roles/bigquery.jobUser`) at the project level. This allows federated users to submit and manage their own query and load jobs within the project.

To set up your Google Workforce Identity Pool:

Run the following command to create a Workforce Identity Pool, replacing `<your-organization-id>` with your Google Cloud Organization ID:

Create a Workforce Identity Pool

```
gcloud iam workforce-pools create my-org-workforce-pool \
  --organization=organizations/<your-organization-id> \
  --display-name="My Organization Workforce Pool" \
  --description="Workforce Identity Pool for my organization"
```

Note the value of `my-org-workforce-pool` as this will be your `your-workforce-pool-id`.

Obtain Dremio's public JWKS and Dremio's issuer from the BigQuery source configuration in Dremio. Retrieve these values by performing one of the following:

- Click the Workforce Identity Federation button in the Dremio console.
- Run these API calls:

Retrieve Dremio's public JWKS by sending a GET request:

Retrieve Dremio's public JWKS

```
/v3/external-oauth/discovery/jwks
```

Save the response as `public_jwk_set.json`. This file contains Dremio's public key set, which Google will use to verify signed JWTs.

Retrieve Dremio's issuer by sending a GET request:

Retrieve Dremio's issuer

```
/v3/external-oauth/discovery/jwt-issuer
```

Note the issuer value returned. You'll need it when configuring Google Cloud to establish the trust relationship.

Run following bash command, replacing ``<Dremio_issuer_value>`` with the value obtained in the previous step:

Create a Workforce identity provider

```
gcloud iam workforce-pools providers create-oidc my-workforce-provider \
  --workforce-pool=my-org-workforce-pool \
  --display-name='My Dremio Provider' \
  --description='Dremio Provider for BQ impersonation' \
  --issuer-uri='<Dremio_issuer_value>' \
  --client-id='dremio-bq-client-id' \
  --web-sso-response-type="id-token" \
  --web-sso-assertion-claims-behavior="only-id-token-claims" \
  --attribute-mapping="google.subject=assertion.sub" \
  --jwk-json-path=./public_jwk_set.json \
  --detailed-audit-logging \
  --location=global \
  --organization=organizations/<your-organization-id>
```

| Parameter | Description |

| --- | --- |

| ``my-workforce-provider`` | The ID you assign to your Workforce identity provider. Note this value for future reference as ``<your-workforce-provider-id>``. |

| ``workforce-pool=my-org-workforce-pool`` | Specifies the Workforce Identity Pool this provider belong to. |

| ``display-name='My Dremio Provider'`` | A human-readable name for the provider, as it appears in the Google Cloud Console. |

| ``description='Dremio Provider for BQ impersonation'`` | A short description of the provider's purpose, for example, used for Dremio ↔ BigQuery impersonation. |

| ``issuer-uri='https://internal-issuer.dremio.com'`` | The issuer (iss) claim expected in the JWT signed by Dremio. This should uniquely identify your Dremio instance or the asserting entity. It doesn't need to be publicly resolvable, but it must be consistent. |

| ``client-id='dremio-bq-client-id'`` | The aud (audience) claim that Google expects in the JWT. For this workflow, this typically represent the Dremio client or the Google resource. |

| ``web-sso-response-type="id-token"`` | Specifies the OIDC response type for SSO flows. "id-token" means the Dremio returns an ID token directly. |

| ``web-sso-assertion-claims-behavior="only-id-token-claims"`` | Controls which claims are included in the assertion. "only-id-token-claims" limits it to claims present in the ID token. |

| ``attribute-mapping="google.subject=assertion.sub"`` | Maps the sub (subject) claim from Dremio's JWT to the google.subject attribute (required). |

| ``jwk-json-path=./public_jwk_set.json`` | Path to Dremio's public JWK file. Google uses this to verify Dremio's JWT signatures. |

| ``detailed-audit-logging`` | Enables detailed logging in Cloud Logging — recommended for troubleshooting. |

| ``location=global`` | Sets the Workforce Identity Pool resource location (typically global). |

| ``organization=organizations/<your-organization-id>`` | Specifies your Google Cloud Organization ID. |

Run the following command to grant IAM policy binding for BigQuery job execution, replacing ``<project-id>`` with the Google Cloud project where your BigQuery data resides:

Grant IAM policy binding for BigQuery job execution

```
gcloud projects add-iam-policy-binding <project-id> \
  --role="roles/bigquery.jobUser" \
  --member='principalSet://iam.googleapis.com/locations/global/workforcePools/my-org-workforce-pool/*'
```

If you are using Google Groups mapping, replace ``<your-gcp-access-group>`` with the mapped group name (for example, "BigQueryUsers"):

Grant IAM policy binding for BigQuery job execution with Google groups mapping

```
gcloud projects add-iam-policy-binding <project-id> \
  --role="roles/bigquery.jobUser" \
  --member="principalSet://iam.googleapis.com/locations/global/workforcePools/my-org-workforce-pool/group/your-gcp-access-group"
```

Dremio Configuration

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select the source.

The new source dialog box appears, which contains the following tabs:

General: Create a name for your database, specify the connection details, and set the authentication.

Advanced Options: (Optional) Set the advanced configuration options for your database.

Reflection Refresh: (Optional) Set a policy to control how often Reflections are refreshed and expired.

Metadata: (Optional) Specify dataset handling and metadata refresh.

Privileges: (Optional) Add privileges for users or roles.

Refer to the following sections for guidance on how to edit each tab.

General

To configure the source connection:

For **Name**, enter the name to identify the database in Dremio. The name cannot

include the following special characters: `/`, `:`, `[`, or `]`.

For **Host**, enter the hostname or IP address for the Google BigQuery source.

For **Port**, enter the Google BigQuery port number. The default port is `443`.

For **Project ID**, specify the Google Cloud Project ID that contains your BigQuery datasets (for example, ``<your-bigquery-project-id>``).

For **Authentication**, choose between **Service Account** or **Workforce Identity Federation**.

a. If you choose **Service Account**, complete the following:

For **Client Email**, enter the client email.

For **Service Account Key**, choose an authentication method:

Dremio: Provide the database password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for the Azure Key Vault secret that stores the Vertica password. The URI format is
`https://<vault_name>.vault.azure.net/secrets/<secret_name>` (for example, `<https://myvault.vault.azure.net/secrets/mysecret>`).

note

To use Azure Key Vault as your application secret store, you must:

Deploy Dremio on Azure.

Complete the Requirements for Authenticating with Azure Key Vault.

It is not necessary to restart the Dremio coordinator when you rotate secrets stored in Azure Key Vault. Read Requirements for Secrets Rotation for more information.

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS console or command line tools.

HashiCorp Vault: Choose the HashiCorp secrets engine you're using from the dropdown menu and enter the secret reference for the password in the correct format in the provided field.

b. If you choose **Workforce Identity Federation**, complete the following:

For **Default User**, enter the default user identifier that Dremio will use to fetch metadata and execute reflection jobs. This user must have sufficient privileges to perform these operations. Example: `dremio-svc-user@yourcompany.com`

For **Audience**, enter the client identifier used by Dremio during authentication (e.g., `value: dremio-bq-client-id`). This must exactly match the `client-id` from Step 2 of prerequisites setup.

For **Client ID**, enter the client identifier used by Dremio during authentication (e.g., `value: dremio-bq-client-id`). This must exactly match the `client-id` from Step 2 of the prerequisites setup. In this configuration, the client ID and audience values are the same.

note

Sources containing a large number of files or tables may take longer to be added. During this time, the source name is grayed out and shows a spinner icon, indicating the source is being added. Once complete, the source becomes accessible.

Advanced Options

Set the advanced configuration options for your database:

Record fetch size: Number of records to fetch at once. Set to `0` to have Dremio automatically decide. By default, this is set to `10`.

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to `8`.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to `60`.

Query timeout (s): The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Enable external authorization plugin: When enabled, authorizes an external plugin.

Connection Properties: Connection properties and values for the data source.

Reflection Refresh

Set the policy that controls how often Reflections are refreshed or expired, using the following options:

Never refresh: Select to prevent automatic Reflection refresh; otherwise, the default is to refresh automatically.

Refresh every: How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if **Never refresh** is selected.

Set refresh schedule: Specify the daily or weekly schedule.

Never expire: Select to prevent Reflections from expiring; otherwise, the default is to expire automatically after the time limit specified in **Expire after**.

Expire after: The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if **Never expire** is selected.

Metadata

Set the following metadata options:

Remove dataset definitions if underlying data is unavailable: Checked by default. If this box is *not* checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

-

Data Discovery: Set the time interval for fetching top-level source object names such as databases and tables. You can choose to set the **Fetch every** frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality. Use these parameters to fetch or expire the metadata:

Fetch mode: Fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to **Only Queried Datasets**.

Fetch every: Set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.

Expire after: Set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.

Privileges

To grant privileges to specific users or roles:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

See [Access Control](#) for additional information about privileges.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

```
`*`, `+`, `-`, `/`, `%`  
`<`, `<=`, `<>`, `=`, `>`, `>=`, `!=`  
`AND`, `NOT`, `OR`, `||`  
`ABS`  
`ACOS`  
`ADD_MONTHS`  
`ASIN`  
`ATAN`  
`ATAN2`  
`AVG`  
`CAST`  
`CEIL`  
`CEILING`  
`CHAR_LENGTH`  
`CHARACTER_LENGTH`
```

`CONCAT`
`COS`
`COT`
`DATE_ADD`
`DATE_SUB`
`DATE_TRUNC_DAY`
`DATE_TRUNC_HOUR`
`DATE_TRUNC_MINUTE`
`DATE_TRUNC_MONTH`
`DATE_TRUNC_YEAR`
`DEGREES`
`E`
`EXP`
`EXTRACT_DAY`
`EXTRACT_DOW`
`EXTRACT_DOY`
`EXTRACT_HOUR`
`EXTRACT_MINUTE`
`EXTRACT_MONTH`
`EXTRACT_QUARTER`
`EXTRACT_SECOND`
`EXTRACT_WEEK`
`EXTRACT_YEAR`
`FLOOR`
`IS DISTINCT FROM`
`IS NOT DISTINCT FROM`
`IS NOT NULL`
`IS NULL`
`LAST_DAY`
`LCASE`
`LEFT`
`LENGTH`
`LIKE`
`LN`
`LOCATE`
`LOG`
`LOG10`
`LOWER`
`LPAD`
`LTRIM`
`MAX`
`MEDIAN`
`MIN`
`MOD`
`MONTH`
`PERCENT_CONT`
`PERCENT_DISC`
`PI`
`POSITION`
`POW`
`POWER`
`RADIANS`
`RAND`
`REPLACE`

`REVERSE`
`RIGHT`
`ROUND`
`RPAD`
`RTRIM`
`SIGN`
`SIN`
`SQRT`
`SUBSTR`
`SUBSTRING`
`SUM`
`TAN`
`TIMESTAMPADD_DAY`
`TIMESTAMPADD_HOUR`
`TIMESTAMPADD_MINUTE`
`TIMESTAMPADD_MONTH`
`TIMESTAMPADD_QUARTER`
`TIMESTAMPADD_SECOND`
`TIMESTAMPADD_YEAR`
`TIMESTAMPDIFF_DAY`
`TIMESTAMPDIFF_HOUR`
`TIMESTAMPDIFF_MINUTE`
`TIMESTAMPDIFF_MONTH`
`TIMESTAMPDIFF_QUARTER`
`TIMESTAMPDIFF_SECOND`
`TIMESTAMPDIFF_WEEK`
`TIMESTAMPDIFF_YEAR`
`TO_DATE`
`TRIM`
`TRUNC`
`TRUNCATE`
`UCASE`
`UPPER`
`YEAR`

Data Source Management

Updating the Source

To update the source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the dropdown.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name.

Click **Save**.

Deleting the Source

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the `ADMIN` role can delete the source.

To delete the source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and click [!The Settings icon](#) to the right.

From the dropdown, select **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Sources containing a large number of files or tables may take longer to be removed. During this time, the source name is grayed out and shows a spinner icon, indicating the source is being removed. Once complete, the source disappears.

Querying the Google BigQuery Source Directly

Dremio users can run pass queries through Dremio to run on your database. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[\[Previous](#)

[Elasticsearch\]\(/current/data-sources/databases/elasticsearch\)](#)[\[Next](#)

[IBM Db2\]\(/current/data-sources/databases/ibm-db2\)](#)

[Requirements](#)

[User Impersonation](#)

[Prerequisites](#)

[Configure Google Workforce Identity Pool](#)

[Dremio Configuration](#)

[General](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Metadata](#)

[Privileges](#)

Predicate Pushdowns

Data Source Management

Updating the Source

Deleting the Source

Querying the Google BigQuery Source Directly

Source:
<https://docs.dremio.com/current/data-sources/databases/elasticsearch>

Version: current [26.x]

On this page

Elasticsearch

This topic describes how to configure Elasticsearch as a source in Dremio.

note

If your organization upgrades to Elasticsearch v7.0+, you will need to remove and re-add it as a source in Dremio.

Compatibility

Supported Versions:

Elasticsearch versions 7.x, 8.x, and 9.x (8.x and 9.x in version 7 compatibility mode)

Dremio Software only

Pushdown Scripting Support: Painless

Metadata Concepts

In order to plan and execute queries, Dremio captures and stores Elastic metadata in Dremio's internal metadata database to efficiently plan and execute queries. This captured metadata is broken into two broad categories:

Dataset Discovery: Names of available Indices, Mappings and Aliases.

This information is required to expose databases and tables in the Dremio UI and BI tool connections.

Dataset Details: Complete information including definition of mapping, sampled schema and shard locations. This information is required to complete a query against a particular table.

Dremio will interact with the ``/_cluster/state/metadata`` api to understand the nature of the objects inside your Elasticsearch install. From this API endpoint, Dremio can learn

metadata about each of these object type. By default, Dataset Discovery has an hourly refresh interval. Additionally, Dataset Details has an hourly refresh interval for Elastic tables that have been queried at least once.

Accessing Objects

The Dremio Elastic Connector is designed to provide a consistent and understandable view of Elastic Indices and Mappings, through the use of a two level hierarchy. In Dremio, these two levels can be thought of as database and table. Elastic Indices and Aliases are exposed as databases and each mapping within those index or alias is exposed as a table.

Dremio also supports exposing data inside Elastic aliases. In Dremio, aliases and indices are not visually distinguished and a user can easily interact with either entity. Additionally, Dremio understands filtered aliases and will correctly apply those filters as part of its operations.

Dremio also allows users access to Elastic's capability to expose synthetic tables through wildcards and comma separated lists. A user can use wildcards in both the name of the database (index) or the name of the table (mapping). This is done by modifying the from clause in a standard SQL query. Once that query is executed, if Elastic recognizes the name, those entities will show up in the product as additional datasets available for query and access (and will be maintained and secured like any other table). If you want to have Dremio forget about those entities, an administrator can use ``ALTER TABLE <TABLE> FORGET METADATA`` to remove those synthetic entities.

Access objects examples

```
SELECT * from elastic."feb*"."big"
SELECT * from elastic."feb,march"."big"
SELECT * from elastic."feb"."big,small"
```

Execution Metadata

When Dremio executes queries against Elastic, it usually parallelizes the query to interact with each shard in your Elastic cluster to move data as quickly as possible back to Dremio. Dremio does this by probing the ``/<indexOrAlias>/_search_shards`` API.

List Promotion Rules

Elastic does not distinguish between scalar and list of scalars but Dremio does. In order to ensure the best possible user experience, Dremio uses the schema analysis phases outlined above to expose the final user schema. To simplify things, once Dremio detects at least one field with a list of scalars, it exposes all records for that field as a list of scalars. This allows users to avoid having to deal with union types. An example:

-

Elastic mapping is defined as field 'A' and type integer.

Records 1-4 exist and each have a single integer for field 'A'.

Dremio samples the schema and exposes field 'A' as a scalar.

Record five is inserted into the index

Dremio now exposes field 'A' as an `int[]` for all records 1-5.

Dremio does this promotion both at initial sampling time and during execution.

If during execution Dremio discovers a value for a field that is of scalar type is actually a list type, Dremio will learn this schema and re-execute the query.

Special Handling for Geoshape

Geoshape is a special type in the Elastic ecosystem.

This is because has a different schema depending on which type is exposed.

Despite this, they are all represented at the type system level as a single type.

In this situation, Dremio exposes the Geoshape type and specifically its potential coordinates fields as a group of union fields supporting from 1 to 4-dimensional double arrays

to reflect the various types of Elastic geoshapes.

Mapping Consistency and Schema Learning

In some cases, it is possible that Dremio will query an index and find a schema change that was previously unknown to Dremio (different type for field or new field). In both cases, Dremio will do a two step verification process to correctly learn the new schema. Dremio maintains a mapping checksum for all identified schemas. When it encounters an unexpected change, it will first verify that the canonical schema from Elastic is consistent with Dremio's previously known mapping. If it is, Dremio will follow its standard promotion rules. If it is not, Dremio will halt execution and request the user to use the `ALTER TABLE <TABLE> REFRESH METADATA`

operation to have Dremio immediately reread the updated Elastic mapping information. Note, this is an optional step as the mapping will also be updated on the schedule defined for automated metadata updates.

Discovery of New Fields

As part of the Dataset Details refresh, Dremio will automatically reload all Elastic mappings to learn about any new fields. Each time this happens, Dremio will resample and update its understanding of schema.

Mapping Merging

If you compose a query that includes multiple mappings, Dremio will do its best to merge those mappings. Mappings are merged on a field by field basis. Mappings can be merged if at least one of the following is true:

Fields with overlapping positions are the same type (`mapping1.a::int`` and `mapping2.a::int``)

Fields are in non-overlapping positions (``mapping1.a::int`` versus ``mapping2.b::float``)

When Dremio merges a mapping, it does so linearly, inheriting the initial field order based on the first index queried.

Elastic Pushdowns

Dremio supports multiple types of pushdowns for different Elastic version and configuration combinations including:

Predicate (e.g. $x < 5$) pushdowns using Elastic queries

Lucene search queries using the ``CONTAINS`` syntax (starting from 5.3.x)

Optional source field/inclusion exclusion (disabled for performance reasons but can be enabled if Dremio has a slow connection to Elastic nodes).

Group by pushdowns for grouping by strings, dates, times, timestamps, integer, longs, doubles, floats, booleans using the Elastic Term Aggregation capabilities

Aggregate Measure pushdown including ``COUNT``, ``COUNT(DISTINCT)``, ``SUM``, ``AVG``, ``STDDEV``, ``VAR`` using Elastic aggregation framework.

Support for converting many arbitrary expressions and ~50 common functions through the use of Groovy (ES2) or Painless (ES5+) scripts for use in both filter and aggregate expressions.

Expression and Function Pushdowns

Dremio supports pushing down the following expressions and functions:

Type	Expression/Function
---	---
Comparison	Equals
Comparison	Not equals
Comparison	Greater than
Comparison	Greater or equal to
Comparison	Less than
Comparison	Less or equal to
Comparison	LIKE
Comparison	ILIKE
Boolean	NOT
Boolean	OR
Boolean	AND
NULL Check	IS NULL
NULL Check	IS NOT NULL
Flow	CASE
Type Conversion	CAST
String	CHAR LENGTH
String	UPPER
String	LOWER
String	TRIM
String	CONCAT
Numeric	Add

Numeric	Subtract
Numeric	Multiply
Numeric	Divide
Numeric	POWER
Numeric	MOD
Numeric	ABS
Numeric	EXP
Numeric	FLOOR
Numeric	CEIL
Numeric	LOG
Numeric	LOG10
Numeric	SQRT
Numeric	SIGN
Numeric	COT
Numeric	ACOS
Numeric	ASIN
Numeric	ATAN
Numeric	DEGREES
Numeric	RADIANS
Numeric	SIN
Numeric	COS
Numeric	TAN

How Dremio Decides What To Pushdown

Dremio works hard to pushdown as many operations as possible to Elastic to try to provide the highest performance experience. Dremio is also focused on maintaining a consistent SQL experience for users who may not understand Elastic or its APIs. As such, Dremio is very focused on providing a correct SQL experience. This includes respecting null semantics through the use of missing aggregation, expression evaluation consistency, correct aggregation semantics on analyzed fields, etc. Dremio also works well with Groovy and Painless to pushdown many more types of operations. It will work without scripts enabled but it is strongly recommended to enable scripts.

Given the nature of Elastic's API, Dremio utilizes the following pieces of functionality to provide a SQL experience: Bucket Aggregations, Pipeline Aggregations, Filter Aggregations and searches using Elastic Query DSL.

Script Construction

Dremio builds custom Groovy (ES2) or Painless (ES5) scripts to interact with Elastic. Because of the many small differences in these languages (type handling, dynamic dispatch, type coercion, function signatures, primitive handling, etc), these scripts are different for each version of Elasticsearch. These scripts utilize Elastic's doc values columnar capability where possible but also rely on `_source` fields for certain operations (e.g. aggregations on analyzed fields for example). As Dremio analyzes a user's SQL expression, it decomposes the expression into a script that can be understood by Elastic's scripting capability.

There are many situations where Dremio uses an expression that might at first be unexpected. These are because of the nature of some of Elastic apis. Some examples behaviors that Dremio does to ensure correct results:

Dremio uses `_source` fields for accessing IP addresses when aggregating or filtering in ES2 because the type has changed between ES2 and ES5

Dremio doesn't push down multi-index complex expressions (``table1.a[2].b[3].c[4]``) using ``doc`` values because doc values can only reference leaf fields and leaf arrays

Dremio doesn't do any array dereferencing using ``_source`` fields because they are not canonicalized to the Elastic mapping. This means that nested arrays ``[1,[2,3]]`` haven't been flattened to the Elastic canonical representation ``[1,2,3]``. This is done as otherwise scripts would produce wrong result.

Dremio won't use a doc field reference for a field that has it implicitly disabled (``string/text``) or explicitly disabled (``doc_values: false``).

Dremio won't use ``doc`` fields for GeoShapes. This is because Dremio doesn't expose a first class shape objects and the fields exposed in Dremio (lists of arrays of doubles) are not directly related to Elastic's internal representation or query capabilities.

Dremio won't pushdown operations against nested fields. This is because nested fields are stored out of line of the core document (not in the original document's doc values) and have semantics inconsistent with traditional SQL aggregation. (Dremio is exploring future work to expose this through enhancements to the language.) Note that Dremio also doesn't use ``_source`` field scripts to interact with nested documents because they are exposed as arrays of values and suffer from the canonicalization issue described above.

Debugging and Logging

If you want to better understand how Dremio is interacting with your Elastic cluster, you can enable Dremio Elastic logging on each Dremio node. This will record each response and request to the Elastic cluster, including a portion of each message body.

You can do this by adding the following configuration to your ``conf/logback.xml`` file on all nodes:

Configuration for `conf/logback.xml` file

```

                                <appender                                name="elasticoutput"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${dremio.log.path}/elastic.log</file>
  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
    <fileNamePattern>${dremio.log.path}/archive/elastic.%d{yyyy-MM-dd}.log.gz
    </fileNamePattern>
    <maxHistory>30</maxHistory>
  </rollingPolicy>
  <encoder>
    <pattern>%date{ISO8601} [%thread] %-5level %logger{36} - %msg%n
    </pattern>
  </encoder>
</appender>
<logger name="elastic.requests" additivity="false">
  <level value="info"/>
  <appender-ref ref="elasticoutput"/>
</logger>
```

Working with Elasticsearch and x-pack

If your Elasticsearch source uses Shield, then your Elasticsearch user account must have the

'monitor' privilege at the cluster level (an admin user has this by default).

In addition, for each index you want to query upon, your user account need to have the 'read' and 'view_index_metadata' privileges as well. Both privileges are included in 'all'.

The following is an example to set up a role 'dremio' with necessary privileges to access

'test_schema_1' index:

Grant privileges to 'dremio' role

```
POST /_xpack/security/role/dremio
```

```
{
  "cluster": [ "monitor" ],
  "indices": [
    {
      "names": [ "test_schema_1" ],
      "privileges": [ "read", "view_index_metadata" ]
    }
  ]
}
```

Working with Elasticsearch and Shield

If your Elasticsearch source uses Shield, then your Elasticsearch user account must have the 'monitor' privilege at the cluster level (an admin user has this by default). If your account lacks the 'monitor' privilege, and you don't have access to an admin user, you can create a new account with 'monitor' by following these steps:

Log in to a search node, go the Elasticsearch install's home directory, and open the file ./config/shield located inside.

Append this text, which gives monitor privileges to an Elasticsearch index called `books` for any user with the `dremio_user` role:

Text to append to ./config/shield file

```
dremio_user:
cluster:
- cluster:monitor/nodes/info
- cluster:monitor/state
- cluster:monitor/health
indices:
'books' :
- read
- indices:monitor/stats
- indices:admin/get
- indices:admin/mappings/get
```

```
- indices:admin/shards/search_shards
```

Run this command, adding a new user to Shield that has the 'dremio_user' role:

Add new user to Shield

```
./bin/shield/esusers useradd <username> -r dremio_user
```

Copy the Shield config file you edited to every other node in the Elasticsearch cluster:

Copy Shield config file to other nodes

```
scp -r ./config/shield root@<other-es-node>:<elastic-install-dir>/config
```

Configuring Elasticsearch as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Elasticsearch**.

General

Name

Specify the name you want to use for the Elasticsearch data source in Dremio. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Connection

Host: Provide the name of the host to use to connect to the Elasticsearch data source.

Port: Provide the port to use with the specified hostname to connect to the Elasticsearch data source (default is `9200`).

Encrypt connection: Select or deselect the checkbox to specify whether Dremio should encrypt the connection to the Elasticsearch data source.

Managed Elasticsearch service: Select the checkbox if you are connecting to a managed Elasticsearch instance or Dremio only has access to the specified host.

Authentication

No Authentication

Master Credentials (default):

Username: Elasticsearch username

Password: Select the password store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Advanced Options

Select or deselect the checkboxes to configure settings for the following options:

Show hidden indices that start with a dot (.)

Use Painless scripting with Elasticsearch 5.0+ (experimental)

Show `_id` columns

Use index/doc fields when pushing down aggregates and filters on analyzed and normalized fields (may produce unexpected results)

Perform keyword searches when pushing down fields mapped as text and keyword

Use scripts for query pushdown

If the number of records returned from Elasticsearch is less than the expected number, warn instead of failing the query

Force Double Precision

Specify the desired settings for the following options:

Read timeout (seconds)

Scroll timeout (seconds)

Scroll size: Setting must be less than or equal to your Elasticsearch setting for ``index.max_result_window``, which typically defaults to 10,000.

Under **Encryption**, choose a Validation Mode option:

Validate certificate and hostname

Validate certificate only

Do not validate certificate or hostname

Reflection Refresh

Refresh Settings: Select whether to never refresh Reflections; refresh at an interval based on hours, days, or weeks; or refresh at the specified schedule.

Expire Settings: Select whether Reflections should never expire or expire at an interval based on hours, days, or weeks.

Metadata

Under **Dataset Handling**, select or deselect the checkbox to specify whether Dremio should remove dataset definitions if underlying data is unavailable.

Under **Metadata Refresh**:

Dataset Discovery: Specify the refresh interval to use for the names of top-level source objects such as tables.

Dataset Details: Specify refresh and expiration intervals for the metadata Dremio needs for query planning, such as information on fields, types, shards, statistics and locality.

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an Elasticsearch Source

To update an Elasticsearch source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Elasticsearch as a Source](#).

Click **Save**.

Deleting an Elasticsearch Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Elasticsearch source, perform these steps:

On the Datasets page, click **Sources** > **Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

For More Information

See [Elasticsearch Data Types](#) for information about mapping to Dremio data types.

Was this page helpful?

[Previous

Dremio Cluster](/current/data-sources/databases/dremio)[Next

Google BigQuery](/current/data-sources/databases/google-bigquery)

Compatibility

Metadata Concepts

Accessing Objects

Execution Metadata

List Promotion Rules

Special Handling for Geoshape

Mapping Consistency and Schema Learning

Discovery of New Fields

Mapping Merging

Elastic Pushdowns

Expression and Function Pushdowns

How Dremio Decides What To Pushdown

Script Construction

Debugging and Logging

Working with Elasticsearch and x-pack

Working with Elasticsearch and Shield

Configuring Elasticsearch as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating an Elasticsearch Source

Deleting an Elasticsearch Source

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/ibm-db2>

Version: current [26.x]

On this page

IBM Db2

You can add Db2 databases as sources to Dremio.

See [IBM Db2 Data Types](#) for information about mapping to Dremio data types.

Limitations

Only IBM Db2 for Linux, UNIX, and Windows is supported.

Configuring IBM Db2 as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **IBM Db2**.

General Options

In the **Name** field, specify the name by which you want the Db2 source to appear in the **Databases** section. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Under **Connection**, follow these steps:

In the **Host** field, specify the hostname or IP address of the database to connect to.

In the **Port** field, specify the port to use when connecting. The default is 50000.

In the **Database** field, specify the name of the database.

Under **Authentication**, specify the Db2 username. Then, choose a method for providing the Db2 password from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Advanced Options

On the Advanced Options page, you can set values for these non-required options:

Option	Description
---	---
Maximum Idle Connections	The total number of connections allowed to be idle at a given time. The default maximum idle connections is 8.
Connection Idle Time	The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. The default connection idle time is 60 seconds.
Query Timeout	The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Reflection Refresh Options

On the Reflection Refresh page, set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata Options

On the Metadata page, you can configure settings to refresh metadata and handle datasets.

Dataset Handling

These are the optional **Dataset Handling** parameters.

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

---	---
-----	-----

Remove dataset definitions if underlying data is unavailable	By default, Dremio removes dataset definitions if underlying data is unavailable. Useful when files are temporarily deleted and added back in the same location with new sets of files.
---	---

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

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

---	---
-----	-----

(Optional) Fetch every	You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.
-------------------------------	---

Dataset Details: The metadata that Dremio needs for query planning such as information required for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

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

---	---
-----	-----

Fetch mode	You can choose to fetch only from queried datasets that are set by default. Dremio updates details for previously queried objects in a source. Fetching from all datasets is deprecated.
-------------------	--

Fetch every	You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.
--------------------	---

Expire after	You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.
---------------------	--

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an IBM Db2 Source

To update an IBM Db2 source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring IBM Db2 as a Source](#).

Click **Save**.

Deleting an IBM Db2 Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an IBM Db2 source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

```
`||`, `AND`, `OR`  
`=`, `+`, `-`, `/`, `*`  
`<=`, `<`, `>`, `>=`, `=` , `<>`, `!=`  
ABS
```

ADD_MONTHS
AVG
BETWEEN
CASE
CAST
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
COALESCE
CONCAT
DATE_ADD
DATE_DIFF
DATE_TRUNC
DATE_TRUNC_CENTURY
DATE_TRUNC_DAY
DATE_TRUNC_DECADE
DATE_TRUNC_HOUR
DATE_TRUNC_MICROSECOND
DATE_TRUNC_MILLENIUM
DATE_TRUNC_MILLISECOND
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER
DATE_TRUNC_SECOND
DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DAYOFMONTH
DAYOFWEEK
DAYOFYEAR
EXTRACT
FLOOR
ILIKE
IN
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LEFT
LENGTH
LIKE
LOCATE
LOWER
LPAD
LTRIM
MAX
MIN
MOD
NOT
OR
PERCENT_RANK
POSITION
REPLACE

RIGHT
ROUND
RPAD
RTRIM
SIGN
SQRT
STDDEV
STDDEV_POP
STDDEV_SAMP
SUBSTR
SUBSTRING
SUM
TO_CHAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UPPER

Running Queries Directly on IBM Db2 Through Dremio

Dremio users can run pass queries through Dremio to run on IBM Db2. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[\[Previous](#)

[Google BigQuery\]\(/current/data-sources/databases/google-bigquery\)\[Next](#)

[Microsoft Azure Data Explorer\]\(/current/data-sources/databases/azure-data-explorer\)](#)

[Limitations](#)

[Configuring IBM Db2 as a Source](#)

[General Options](#)

[Advanced Options](#)

[Reflection Refresh Options](#)

[Metadata Options](#)

[Privileges](#)

[Updating an IBM Db2 Source](#)

[Deleting an IBM Db2 Source](#)

[Predicate Pushdowns](#)

[Running Queries Directly on IBM Db2 Through Dremio](#)

Source:
<https://docs.dremio.com/current/data-sources/databases/azure-data-explorer>

Version: current [26.x]

On this page

Microsoft Azure Data Explorer

You can add a source to Dremio that is a database in [Azure Data Explorer \(ADX\)](#).

Prerequisites

Ensure that you have the URI for connecting to the ADX cluster in which the database is located.

Ensure that you know the name of the database that you want to add as a source.

Configuring Azure Data Explorer as a Source

To add a database that is in an ADX cluster as a source in Dremio:

Click the Settings icon in the left navigation bar and select **Support**.

In the **Support Keys** section of the Support Settings page, add the support key ``plugins.jdbc.adx.enabled`` and toggle it on.

Navigate to the Datasets page. To the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Microsoft Azure Data Explorer**.

General

Under **Name**, enter the name to use for the Azure Data Explorer source. The name cannot include the following special characters: ``/``, ``:``, ``[``, or ``]``.

Connection

Describe the Data Explorer cluster used in this connection.

Cluster URI: Enter the cluster URI.

Tenant ID: Enter the directory (tenant) ID.

Authentication

Select an authentication option:

Microsoft Entra ID: To register a Microsoft Entra ID application and obtain the required IDs and client secret, see [How to register an app in Microsoft Entra ID](#), then complete the Dremio configuration:

Application ID: Enter the application (client) ID

Application Secret: Select the application secret store from the dropdown menu:

Dremio: Provide the application secret in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the application secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the secret reference in the required format.

Azure Managed Identity: Passwordless authentication using Azure's managed identity service, eliminating credential management overhead.

Create a managed identity, system-assigned or user-assigned for your Data Explorer cluster.

Attach the managed identity to your Dremio AKS cluster's Virtual Machine Scale Set (VMSS)

Configure the Dremio source:

When using a user-assigned managed identity, add the **Client ID** to the Dremio source configuration.

When using a system-assigned managed identity, leave the **Client ID** blank.

Under **Database Name**, enter the name of the database that you want to add as a source. Names are case-sensitive.

Advanced Options

On the Advanced Options page, you can set values for these non-required options:

Option	Description
--------	-------------

---	---
-----	-----

Maximum Idle Connections	The total number of connections allowed to be idle at a given time. The default maximum idle connections is 8.
---------------------------------	--

Connection Idle Time	The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. The default connection idle time is 60 seconds.
-----------------------------	--

Query Timeout	The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.
----------------------	---

Reflection Refresh Options

On the Reflection Refresh page, set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata Options

On the Metadata page, you can configure settings to refresh metadata and handle datasets.

Dataset Handling

These are the optional **Dataset Handling** parameters.

Parameter	Description
---	---
Remove dataset definitions if underlying data is unavailable	By default, Dremio removes dataset definitions if underlying data is unavailable. Useful when files are temporarily deleted and added back in the same location with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

Parameter	Description
---	---
(Optional) Fetch every	You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning such as information required for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

Parameter	Description
---	---
Fetch mode	You can choose to fetch only from queried datasets that are set by default. Dremio updates details for previously queried objects in a source. Fetching from all datasets is deprecated.
Fetch every	You can choose to set the frequency to fetch dataset details in

minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour. |
| **Expire after** | You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours. |

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating an ADX Source

To update an ADX source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Azure Data Explorer as a Source](#).

Click **Save**.

Deleting an ADX Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an ADX source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the ADX source and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Query Pushdowns

Dremio can delegate the execution of these expressions and functions to the database being queried, often dramatically improving query performance.

`-`,`=`,`+`,`*`,`/`,`<`,`<=`,`<>`,`>`,`>=`,`>`,`!=`,`

ADD_MONTHS

AND, LIKE, NOT, OR, ||

AVG

CAST

CONCAT

COUNT *

COUNT DISTINCT

COUNT DISTINCT MULTI

COUNT MULTI

DATE_ADD

DATE_SUB

DATE_TRUNC_DAY

DATE_TRUNC_HOUR

DATE_TRUNC_MINUTE

DATE_TRUNC_MONTH

DATE_TRUNC_YEAR

FULL JOIN

INNER JOIN

IS DISTINCT FROM

IS NOT DISTINCT FROM

IS NOT NULL

IS NULL

JOIN

LAST_DAY

LEFT JOIN

MAX

MIN

MOD

RIGHT JOIN

SIGN

SORT

SUM

Running Queries Directly on Azure Data Explorer Through Dremio

Dremio users can run pass queries through Dremio to run on Azure Data Explorer. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[Previous

IBM Db2](/current/data-sources/databases/ibm-db2)[Next

Microsoft Azure Synapse Analytics](/current/data-sources/databases/azure-synapse-analytics)

Prerequisites

Configuring Azure Data Explorer as a Source

General

Advanced Options

Reflection Refresh Options

Metadata Options

Privileges

Updating an ADX Source

Deleting an ADX Source

Query Pushdowns

Running Queries Directly on Azure Data Explorer Through Dremio

Source:
<https://docs.dremio.com/current/data-sources/databases/azure-synapse-analytics>

Version: current [26.x]

On this page

Microsoft Azure Synapse Analytics

Dremio supports integrations with organizations using Azure Synapse Analytics dedicated SQL pools via the external source.

Requirements

Dremio v19.3+

Configuring Synapse Analytics as a Source

On the Datasets page, to the right of **Sources** in the left panel, click !This is the Add Source icon.

In the Add Data Source dialog, under **Databases**, select **Microsoft Azure Synapse Analytics**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Connection

Describe the Synapse Analytics SQL Server workspace used in this connection.

Under **Host**, enter the URL for your dedicated SQL pool, which typically ends in `.sql.azure-synapse.net``.

Under **Port (optional)**, enter the port required to access the data source.

Under **Database**, enter the database's name. Only this database is accessed by Dremio.

Authentication

Select an authentication option:

No Authentication: Dremio does not attempt to provide any authentication when connecting with the SQL pool.

Master Credentials: Dremio must provide a specified username and password in order to access the SQL pool.

Username: Enter the Microsoft Azure Synapse Analytics username.

Password: Select the password store from the dropdown menu:

Dremio: Provide the Microsoft Azure Synapse Analytics password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the Microsoft SQL Server password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Microsoft Entra ID: To register a Microsoft Entra ID application and obtain the required IDs and client secret, see [How to register an app in Microsoft Entra ID](#), then complete the Dremio configuration:

Tenant ID: Unique identifier of your Microsoft Entra ID tenant.

Application ID: Enter the application (client) ID

Application Secret: Select the application secret store from the dropdown menu:

Dremio: Provide the application secret in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the application secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the secret reference in the required format.

Azure Managed Identity: Passwordless authentication using Azure's managed identity service, eliminating credential management overhead.

Create a managed identity, system-assigned or user-assigned for your Synapse workspace.

Attach the managed identity to your Dremio AKS cluster's Virtual Machine Scale Set (VMSS)

Configure the Dremio source:

When using a user-assigned managed identity, add the **Client ID** to the Dremio source configuration.

When using a system-assigned managed identity, leave the **Client ID** blank.

Select the **Encrypt connection** option to encrypt the connection to Microsoft Azure Synapse Analytics. Clear the checkbox to disable encryption.

Advanced Options

The following settings control more advanced functionalities in Dremio.

Advanced Options

Show only the initial database used for connecting - This restricts Dremio's access only to a default database as specified on the **General** table.

Record fetch size - Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. The default value is ``10``.

Maximum idle connections - The maximum number of idle connections to keep.

Connection idle time (s) - Idle time, in seconds, before a connection is considered for closure.

Query timeout (s) - The timeout, in seconds, for query execution before it is canceled. Set to ``0`` for no timeout.

Encryption

Verify server certificate - Forces Dremio to verify the server's certificate using the distinguished name.

SSL/TLS server certificate distinguished name - Specifies the location for the certificate server, which must be set to ``*.sql.azure-synapse.net``.

Connection Properties

Name - The unique name for any custom properties.

Value - The value associated with the custom property.

Reflection Refresh

This tab controls the frequency of Reflection refreshes or the timespan for expiration for any queries performed using this data source.

Never refresh - Prevents any query Reflections associated with this source from refreshing.

Refresh every - Sets the time interval by which Reflections for this source are refreshed. This may be set to hours, days, and weeks.

Never expire - Prevents any query Reflections associated with this source from expiring.

Expire after - Sets the time after a Reflection is created that it then expires and can no longer be used for queries. This may be set to hours, days, and weeks.

Metadata

This tab offers settings that control how dataset details are fetched and refreshed.

Dataset Handling

Remove dataset definitions if underlying data is unavailable - If this box is not checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery

Fetch every - Specifies the time interval by which Dremio fetches object names. This can be set by minutes, hours, days, and weeks.

Dataset Details

Fetch mode - Restricts when metadata is retrieved.

Fetch every - Specifies the time interval by which metadata is fetched. This can be set by minutes, hours, days, and weeks.

Expire after - Specifies the timespan for when dataset details expire after a dataset is queried. This can be set by minutes, hours, days, and weeks.

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access](#)

Controls for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Microsoft Azure Synapse Analytics Source

To update a Microsoft Azure Synapse Analytics source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the !The Settings icon at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Microsoft Azure Synapse Analytics as a Source](#).

Click **Save**.

Deleting a Microsoft Azure Synapse Analytics Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Microsoft Azure Synapse Analytics source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

`%`,`*`,`+`,`-`,`/`
`<`,`<=`,`<>`,`=`,`>`,`>=`,`!=`
AND, NOT, OR, `||`
ABS
ACOS
ADD_MONTHS
ASIN
ATAN
ATAN2
AVG
CAST
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COT
DATE_ADD
DATE_SUB
DATE_TRUNC_DAY
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_YEAR
DEGREES
E
EXP
EXTRACT_DAY
EXTRACT_DOW
EXTRACT_DOY
EXTRACT_HOUR
EXTRACT_MINUTE
EXTRACT_MONTH
EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE

LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MIN
MOD
MONTH
PI
POSITION
POW
POWER
RADIANS
RAND
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
SUBSTR
SUBSTRING
SUM
TAN
TIMESTAMPADD\ _DAY
TIMESTAMPADD\ _HOUR
TIMESTAMPADD\ _MINUTE
TIMESTAMPADD\ _MONTH
TIMESTAMPADD\ _QUARTER
TIMESTAMPADD\ _SECOND
TIMESTAMPADD\ _YEAR
TIMESTAMPDIFF\ _DAY
TIMESTAMPDIFF\ _HOUR
TIMESTAMPDIFF\ _MINUTE
TIMESTAMPDIFF\ _MONTH
TIMESTAMPDIFF\ _QUARTER
TIMESTAMPDIFF\ _SECOND
TIMESTAMPDIFF\ _WEEK
TIMESTAMPDIFF\ _YEAR
TO\ _DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
YEAR

Running Queries Directly on Azure Synapse
Analytics Through Dremio

Dremio users can pass queries through Dremio to run on Azure Synapse Analytics. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[Previous

Microsoft Azure Data Explorer](/current/data-sources/databases/azure-data-explorer)[Next

Microsoft SQL Server](/current/data-sources/databases/sql-server)

Requirements

Configuring Synapse Analytics as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Microsoft Azure Synapse Analytics Source

Deleting a Microsoft Azure Synapse Analytics Source

Predicate Pushdowns

Running Queries Directly on Azure Synapse Analytics Through Dremio

Source:
<https://docs.dremio.com/current/data-sources/databases/sql-server>

Version: current [26.x]

On this page

Microsoft SQL Server

This topic provides Microsoft SQL Server data source setup and configuration information.

Supported Versions

Dremio supports Microsoft SQL Server 2012 and later.

note

Ensure that your Dremio cluster has access to the appropriate port for your Microsoft

SQL Server source. By default, this is port 1433.

Initial Connection

Depending on the number of tables in your SQL Server source, the final step of adding it to Dremio can take anywhere from a few seconds to a few minutes as the source's metadata is processed. However, this is a one-time cost and further queries to the source will not incur additional metadata reads.

User Impersonation

The Microsoft SQL Server username provided in the source configuration is the default username that is used for running queries. When queries are run against Microsoft SQL Server in Dremio, users use the privileges associated with the Microsoft SQL Server username and run queries under that username.

You can change this default in Dremio by enabling user impersonation in the Advanced Options, which allows users to run queries under their own usernames and restricts their access. For example, ``user_1`` can run queries as ``user_1`` rather than ``sqlsvr_svc``. Before enabling user impersonation, some setup is required in Microsoft SQL Server to allow one user to impersonate another user because the username of the user in Dremio must be the same as their username in Microsoft SQL Server and the user must be able to connect through the Microsoft SQL Server username.

note

Reflections are not supported on data sources with user impersonation enabled to ensure that all security and governance policies defined in the underlying data source are enforced.

Reflections created prior to enabling user impersonation must be manually dropped, as they will fail to refresh once impersonation is active.

To set up user impersonation, follow these steps:

Ensure the user's username in Microsoft SQL Server matches their username in Dremio. If the usernames do not match, modify one of the usernames or create a new user account with a matching username.

Run a GRANT IMPERSONATE command in Microsoft SQL Server to allow the user to connect through their Microsoft SQL Server username:

Example of granting impersonate privilege in Microsoft SQL Server

```
GRANT IMPERSONATE ON USER::testuser1 TO proxyuser;
```

In this example, the user can log in as ``testuser1`` in Dremio and in Microsoft SQL Server, and they can connect through the ``proxyuser``. The ``proxyuser`` is the Microsoft SQL Server username provided in the source configuration.

Log in to Dremio as a member of the ADMIN role.

Follow the steps for Configuring Microsoft SQL Server as a Source using the Microsoft

SQL Server username `proxyuser` and enable **User Impersonation** in the **Advanced Options**.

Grant source privileges to the user.

Now that you have enabled user impersonation, a user who logs in to Dremio with their username can access the Microsoft SQL Server source and its datasets according to their privileges. The user can also run queries against Microsoft SQL Server under their username.

Configuring Microsoft SQL Server as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Microsoft SQL Server**.

General

Name

Enter the name to identify the data source in Dremio. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Connection

Describe the SQL Server instance used in this connection.

Host: The SQL Server host name or IP address.

Port (Optional): The SQL Server port number. If you do not specify a port number, the SQL Server instance is queried to retrieve the port that the named instance is listening on.

Database (Optional): The database instance name

Authentication

Select an authentication option:

No Authentication: Connects without credentials. Only use when the SQL Server allows anonymous connections or when network-level security controls access.

Master Credentials (default):

Username: Enter the Microsoft SQL Server username

Secret Store: Select the password secret store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format

`https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the secret reference in the required format.

Microsoft Entra ID: To register a Microsoft Entra ID application and obtain the required IDs and client secret, see How to register an app in Microsoft Entra ID, then complete the Dremio configuration:

Tenant ID: Enter the unique identifier of your Microsoft Entra ID tenant.

Application ID: Enter the application (client) ID

Application Secret: Select the application secret store from the dropdown menu:

Dremio: Provide the application secret in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the application secret, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the secret reference in the required format.

Azure Managed Identity (Azure-hosted SQL Server): Passwordless authentication using Azure's managed identity service, eliminating credential management overhead.

Create a managed identity, system-assigned or user-assigned for your SQL Server.

Attach the managed identity to your Dremio AKS cluster's Virtual Machine Scale Set (VMSS)

Configure the Dremio source:

When using a user-assigned managed identity, add the **Client ID** to the Dremio source configuration.

When using a system-assigned managed identity, leave the **Client ID** blank.

Select the **Encrypt connection** option to encrypt the connection to Microsoft SQL Server. Clear the checkbox to disable encryption.

Advanced Options

Show only the initial database used for connecting:

If selected, hides the other DBs that the credential has access to.

Record fetch size: Number of records to fetch at once.

Set to 0 (zero) to have Dremio automatically decide. Default: 10

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to *8*.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to *60*.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Enable legacy dialect

Reflection Refresh

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is *not* checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Microsoft SQL Server Source

To update a Microsoft SQL Server source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Microsoft SQL Server as a Source](#).

Click **Save**.

Deleting a Microsoft SQL Server Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Microsoft SQL Server source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

`*`,`+`,`-`,`/`,`%`
`<`,`<=`,`<>`,`=`,`>`,`>=`,`!=`
AND, NOT, OR, `||`

ABS
ACOS
ADD_MONTHS
ASIN
ATAN
ATAN2
AVG
CAST
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COT
DATE_ADD
DATE_SUB
DATE_TRUNC_DAY
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_YEAR
DEGREES
E
EXP
EXTRACT_DAY
EXTRACT_DOW
EXTRACT_DOY
EXTRACT_HOUR
EXTRACT_MINUTE
EXTRACT_MONTH
EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR

FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE
LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MEDIAN
MIN
MOD
MONTH
PERCENT_CONT
PERCENT_DISC
PI
POSITION
POW
POWER
RADIANS
RAND
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
SUBSTR
SUBSTRING
SUM
TAN
TIMESTAMPADD_DAY
TIMESTAMPADD_HOUR
TIMESTAMPADD_MINUTE
TIMESTAMPADD_MONTH
TIMESTAMPADD_QUARTER
TIMESTAMPADD_SECOND
TIMESTAMPADD_YEAR
TIMESTAMPDIFF_DAY
TIMESTAMPDIFF_HOUR
TIMESTAMPDIFF_MINUTE
TIMESTAMPDIFF_MONTH

TIMESTAMPDIFF_QUARTER
TIMESTAMPDIFF_SECOND
TIMESTAMPDIFF_WEEK
TIMESTAMPDIFF_YEAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
YEAR

note

Since Microsoft SQL Server has no Boolean type, project operations that contain SQL expressions which evaluate to true or false (e.g., `SELECT username, friends > 0`), and filter operations that include Boolean literals in a filter (e.g., `WHERE currentAccount = true`) cannot be executed as pushdowns.

Running Queries Directly on SQL Server Through Dremio

Dremio users can pass queries through Dremio to run on SQL Server. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

For More Information

See [Microsoft SQL Server Data Types](#) for information about mapping to Dremio data types.

Was this page helpful?

[Previous

Microsoft Analytics](/current/data-sources/databases/azure-synapse-analytics)[Next

Azure

Synapse

MongoDB](/current/data-sources/databases/mongo)

Supported Versions

Initial Connection

User Impersonation

Configuring Microsoft SQL Server as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Microsoft SQL Server Source

Deleting a Microsoft SQL Server Source

Predicate Pushdowns

Running Queries Directly on SQL Server Through Dremio

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/mongo>

Version: current [26.x]

On this page

MongoDB

Requirements

To connect to MongoDB, you need:

MongoDB (Dremio supports MongoDB 6.0+)

Access to execute the ``dbStats`` command

Limitation

DX-29932

Queries that un-nest nested fields are not allowed as they would cause incorrect schemas. This may be easily circumvented by pushing filters into the subquery or by simply not referencing the alias.

Configuring MongoDB as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **MongoDB**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ``/``, ``:``, ``[``, or ``\``.

Connection

| Name | Description |

| --- | --- |

| Hosts | A list of Mongo hosts. If MongoDB is sharded, enter the mongos hosts. Otherwise, enter the mongod host. |

| Port | A list of Mongo port numbers. Defaults to 27017. |

Connection Scheme Select how to connect to the source.

Encrypt connection Forces an encrypted connection over SSL.

Read from secondaries only Disables reading from primaries. Might degrade performance.

Authentication

No Authentication

Master Credentials (default):

Username: MongoDB username

Password: Select the password store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Authentication database: Provide the name of the database that Dremio should authenticate against.

Advanced Options

 !

Subpartition Size Number of records to be read by query fragments. This option can be used to increase query parallelism.

Sample Size Number of records to be read when sampling to determine the schema for a collection. If zero the sample size is unlimited.

Sample Method The method (First or Last) by which records should be read when sampling a collection to determine the schema.

Auth Timeout (millis) Authentication timeout in milliseconds.

-

Field names are case insensitive When enabled, Dremio reads all known variations of a field name when determining the schema, ignoring any value set for Sample Size. All field name variations are then used when pushing an operation down to Mongo.

Connection Properties A list of additional MongoDB connection parameters.

Reflection Refresh

 !

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

!

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a MongoDB Source

To update a MongoDB source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring MongoDB as a Source](#).

Click **Save**.

Deleting a MongoDB Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a MongoDB source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio offloads these operations to MongoDB:

ABS
ADD
AND
CASE
CEIL
CONCAT
DAY_OF_MONTH
DIVIDE
EQUAL
EXP
FLOOR
GREATER
GREATER_OR_EQUAL
HOUR
LESS
LESS_OR_EQUAL
LN
LOG
LOG10
MAX
MIN
MINUTE
MOD
MONTH
MULTIPLY
NOT
NOT_EQUAL
OR
POW
REGEX
SECOND
SQRT
SUBSTR
SUBTRACT
TO_LOWER
TO_UPPER
TRUNC
YEAR

For More Information

See [MongoDB Data Types](#) for information about mapping to Dremio data types.

Was this page helpful?

[Previous

Microsoft SQL Server](/current/data-sources/databases/sql-server)[Next

MySQL](/current/data-sources/databases/mysql)

Requirements

Limitation

Configuring MongoDB as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a MongoDB Source

Deleting a MongoDB Source

Predicate Pushdowns

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/mysql>

Version: current [26.x]

On this page

MySQL

Supported Versions

MySQL versions that are 5.5.3 or higher

Configuring MySQL as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **MySQL**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ``/`, `:`, `[`, or `]`.

Host

Name	Description
Host	MySQL host name.
Port	MySQL port number. Defaults to 3306.

Authentication

No Authentication

Master Credentials (default):

Username: MySQL user name

Password secret store:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Store the password securely using URI format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Choose the HashiCorp secrets engine you're using from the dropdown menu and enter the secret reference for the password in the correct format in the provided field.

Advanced Options

![[/assets/images/mysql-adv-options-3887b0b6f7e688153ad2e281536b78ce.png] !

Net write timeout (in seconds): Seconds to wait for data from the server before aborting the connection. Default: 60

Record fetch size: Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. Default: 10

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to *8*.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to *60*.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Connection Properties: Connection properties and values for the data source. If you enable `require_secure_transport` in MySQL, you must add the connection properties `useSSL` and `trustServerCertificate` and set both to the value `true` to prevent errors.

Reflection Refresh

 !

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

 !

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

-

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a MySQL Source

To update a MySQL source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring MySQL as a Source](#).

Click **Save**.

Deleting a MySQL Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a MySQL source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

-

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

`*`,`+`,`-`,`/`,`%`,`<`,`<=`,`<>`,`=`,`>`,`>=`,`!=`,`AND`,`NOT`,`NOT LIKE`,`OR`,`||`

ABS

ACOS

ADD_MONTHS

ASIN

ATAN

ATAN2

AVG

CAST

CEIL

CEILING

CHAR_LENGTH

CHARACTER_LENGTH

CONCAT

COS

COT

CURRENT_DATE

CURRENT_TIME

CURRENT_TIMESTAMP

DATE_ADD

DATE_SUB

DATE_TRUNC_DAY

DATE_TRUNC_DECADE

DATE_TRUNC_HOUR

DATE_TRUNC_MINUTE

DATE_TRUNC_MONTH

DATE_TRUNC_SECOND

DATE_TRUNC_WEEK

DATE_TRUNC_YEAR

DEGREES

E

EXP

EXTRACT_DAY

EXTRACT_DOW

EXTRACT_DOY

EXTRACT_HOUR

EXTRACT_MINUTE

EXTRACT_MONTH

EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE
LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MIN
MOD
PI
POSITION
POW
POWER
RADIANS
RAND
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
STDDEV
STDDEV_POP
STDDEV_SAMP
SUBSTR
SUBSTRING
SUM
TAN
TIMESTAMPADD_DAY
TIMESTAMPADD_HOUR
TIMESTAMPADD_MINUTE
TIMESTAMPADD_MONTH
TIMESTAMPADD_QUARTER
TIMESTAMPADD_SECOND
TIMESTAMPADD_YEAR
TIMESTAMPDIFF_DAY

TIMESTAMPDIFF__HOUR
TIMESTAMPDIFF__MINUTE
TIMESTAMPDIFF__MONTH
TIMESTAMPDIFF__QUARTER
TIMESTAMPDIFF__SECOND
TIMESTAMPDIFF__WEEK
TIMESTAMPDIFF__YEAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
VAR_POP
VAR_SAMP

Running Queries Directly on MySQL Through Dremio

Dremio users can run pass queries through Dremio to run on MySQL. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

For More Information

See [MySQL Data Types](#) for information about mapping to Dremio data types.

Was this page helpful?

[Previous

MongoDB](/current/data-sources/databases/mongo)[Next

Oracle](/current/data-sources/databases/oracle)

Configuring MySQL as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a MySQL Source

Deleting a MySQL Source

Predicate Pushdowns

Running Queries Directly on MySQL Through Dremio

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/oracle>

Version: current [26.x]

On this page

Oracle

This topic describes Oracle data source considerations and Dremio configuration.

User Impersonation

The Oracle database username provided in the source configuration is the default username that is used for running queries. When queries are run against Oracle in Dremio, users use the privileges associated with the Oracle database username and run queries under that username.

You can change this default in Dremio by enabling user impersonation in the Advanced Options, which allows users to run queries under their own usernames and restricts their access. For example, ``user_1`` can run queries as ``user_1`` rather than ``oracle_svc``. Before enabling user impersonation, some setup is required in Oracle to allow one user to impersonate another user, because the username of the user in Dremio must be the same as their username in Oracle and the user must be able to connect through the Oracle database username.

note

Reflections are not supported on data sources with user impersonation enabled to ensure that all security and governance policies defined in the underlying data source are enforced.

Reflections created prior to enabling user impersonation must be manually dropped, as they will fail to refresh once impersonation is active.

To set up user impersonation, follow these steps:

Ensure the user's username in Oracle matches their username in Dremio. If the usernames do not match, modify one of the usernames or create a new user account with a matching username.

Run a `ALTER USER` command in Oracle to allow the user to connect through the Oracle database username:

Example of altering the user in Oracle

```
ALTER USER testuser1 GRANT CONNECT THROUGH proxyuser;
```

In this example, the user can log in as ``testuser1`` in Dremio and in Oracle, and they can connect through the ``proxyuser``. The ``proxyuser`` is the Oracle database username provided in the source configuration.

Log in as an admin to Dremio.

Follow the steps for Dremio Configuration using the Oracle database username and enable **User Impersonation** in the **Advanced Options**.

Grant source privileges to the user.

Now that you have enabled user impersonation, a user logging in to Dremio with their username can access the Oracle source and its datasets according to their privileges. The user also runs queries against Oracle under their username.

Connection Information

The following connection information is needed prior to adding Oracle as a data source.

Hostname or IP

Port

Site Identifier (SID) of the Oracle server

note

Ensure that your Dremio cluster has access to the appropriate port for your Oracle source.
By default this is port 1521.

Initial Connection

Depending on the number of tables in your Oracle source, the final step of adding it to Dremio can take anywhere from a few seconds to a few minutes as the source's metadata is processed. However, this is a one-time cost and further queries to the source will not incur additional metadata reads.

Configuring Oracle as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Oracle**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Host

Name	Description
---	---
Host	Oracle host name.
Port	Oracle port number. Defaults to 1521.
Service Name	Service Name of your database.

| Encrypt connection | Enables secure connections. |

Authentication

Select an authentication option:

No Authentication

Master Credentials (default):

Username: Oracle username

Password: Select the password store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored secret using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Secret Resource Url: Provide the username and secret resource URL for Dremio to use for the source.

Kerberos

Advanced Options

 !

Use timezone as connection region: If checked, uses timezone to set connection region.

Include synonyms: If checked, includes synonyms as datasets.

Map Oracle DATE columns to TIMESTAMP: If selected, the DATE column will display values in timestamp format.

Record fetch size: Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. Default: 10

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to *8*.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to *60*.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Enable legacy dialect

Encryption: Provide the **SSL/TLS server certificate distinguished name**, otherwise,
leave blank to disable the DN match.

Reflection Refresh

 !

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

 !

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery: Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.

This mode minimized metadata queries on a source when not used, but might lead to longer planning times.

-

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Oracle TLS Configuration

To use TLS to connect to an Oracle source, do the following:

Select the option **Connect using SSL/TLS** during initial configuration.

If you want to ensure that the server you're connecting to exactly matches a particular certificate string, add the Server Certificate Distinguished Name under **SSL/TLS** in **Advanced Options**.

Add the Certificate Authority certificate to Dremio's trust store. To add the CA certificate that is used to sign the Oracle certificate into Dremio's trust store:

Import the CA certificate and convert the certificate into DER format (required by Java keytool).

For example, using OpenSSL tool:

```
`$ openssl x509 -outform der -in oracle-ca.pem -out oracle-ca.der`
```

Add the certificate to a new or existing truststore.

```
`$ keytool -import -alias oracle-ca -keystore dremio-truststore.jks -file oracle-ca.der`
```

Modify the ``DREMIO_JAVA_SERVER_EXTRA_OPTS`` section of the ``dremio-env`` configuration

file to use the trust store by adding the following:

```
`Djavax.net.ssl.trustStore=<path/to>/dremio-truststore.jks`
```

```
`Djavax.net.ssl.trustStoreType=JKS`
```

```
`Djavax.net.ssl.trustStorePassword=<password>`
```

Updating an Oracle Source

To update an Oracle source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Oracle as a Source](#).

Click **Save**.

Deleting an Oracle Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete an Oracle source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

```
`*`, `+`, `-`, `/`, `%`  
`<`, `<=`, `<>`, `=`, `>`, `>=`, `!=`  
AND, NOT, OR, `||`  
ABS  
ACOS  
ADD_MONTHS  
ASIN  
ATAN  
ATAN2  
AVG  
CAST  
CEIL  
CEILING
```

CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COSH
COT
COVAR_POP
COVAR_SAMP
DATE_ADD
DATE_SUB
DATE_TRUNC_DAY
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER
DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DEGREES
E
EXP
EXTRACT_CENTURY
EXTRACT_DAY
EXTRACT_DOW
EXTRACT_DOY
EXTRACT_HOUR
EXTRACT_MINUTE
EXTRACT_MONTH
EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE
LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MEDIAN
MIN
MOD
PERCENT_CONT
PERCENT_DISC

PI
POSITION
POW
POWER
RADIANS
REGEXP_LIKE
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SINH
SQRT
STDDEV
STDDEV_POP
STDDEV_SAMP
SUBSTR
SUBSTRING
SUM
TAN
TANH
TO_CHAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
VAR_POP
VAR_SAMP

note

Since Oracle has no Boolean type, project operations that contain SQL expressions which evaluate to true or false (e.g. `SELECT username, friends > 0`), and filter operations that include boolean literals in a filter (e.g. `WHERE currentAccount = true`) cannot be executed as pushdowns.

Running Queries Directly on Oracle Through Dremio

Dremio users can run pass queries through Dremio to run on Oracle. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

For More Information

See [Oracle Data Types](#)
for information about mapping to Dremio data types.

Was this page helpful?

[Previous

MySQL](/current/data-sources/databases/mysql)[Next

PostgreSQL](/current/data-sources/databases/postgres)

User Impersonation

Connection Information

Initial Connection

Configuring Oracle as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Oracle TLS Configuration

Updating an Oracle Source

Deleting an Oracle Source

Predicate Pushdowns

Running Queries Directly on Oracle Through Dremio

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/postgres>

Version: current [26.x]

On this page

PostgreSQL

Configuring PostgreSQL as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **PostgreSQL**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Connection

Name	Description
---	---
Host	Postgres host name.
Port	Postgres port number. Defaults to 5432.
Database	Database name.
Encrypt connection	Enables encrypted connections to Postgres using SSL. Encryption validation mode can be modified under Advanced Options.

Authentication

Select an authentication option:

No Authentication

Master Credentials (default):

Username: PostgreSQL username

Password: Select the password store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format `https://<vault_name>.vault.azure.net/secrets/<secret_name>`

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Secret Resource Url: Provide the username and secret resource URL for Dremio to use for the source.

Advanced Options

Record Fetch Size: Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. By default, this is set to *10*.

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to *8*.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to *60*.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

-

Enable legacy dialect

Encryption

Validation modes include:

Validate certificate and hostname (default)

Validate certificate only

Do not validate certificate or hostname

Reflection Refresh

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.

This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.
This mode minimized metadata queries on a source when not used,
but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a PostgreSQL Source

To update a PostgreSQL source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring PostgreSQL as a Source](#).

Click **Save**.

Deleting a PostgreSQL Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a PostgreSQL source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and

right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

``*`,`+`,`-`,`/`,`%`,`<`,`<=`,`<>`,`=`,`>`,`>=`,`!=`,`AND`,`NOT`,`OR`,`||``
ABS
ACOS
ADD_MONTHS
ASIN
ATAN
ATAN2
AVG
CAST
CBRT
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COT
COVAR_POP
COVAR_SAMP
DATE_ADD
DATE_SUB
DATE_TRUNC_CENTURY
DATE_TRUNC_DAY
DATE_TRUNC_DECADE
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER
DATE_TRUNC_SECOND
DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DEGREES
E
EXP

EXTRACT\ _CENTURY
EXTRACT\ _DAY
EXTRACT\ _DECADE
EXTRACT\ _DOW
EXTRACT\ _DOY
EXTRACT\ _EPOCH
EXTRACT\ _HOUR
EXTRACT\ _MILLENNIUM
EXTRACT\ _MINUTE
EXTRACT\ _MONTH
EXTRACT\ _QUARTER
EXTRACT\ _SECOND
EXTRACT\ _WEEK
EXTRACT\ _YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST\ _DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE
LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MEDIAN
MIN
MOD
PERCENT\ _CONT
PERCENT\ _DISC
PI
POSITION
POW
POWER
RADIANS
REGEXP\ _LIKE
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
STDDEV
STDDEV\ _POP

STDDEV_SAMP
SUBSTR
SUBSTRING
SUM
TAN
TO_CHAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
VAR_POP
VAR_SAMP

Running Queries Directly on PostgreSQL Through Dremio

Dremio users can run pass queries through Dremio to run on PostgreSQL. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

For More Information

See [Postgres Data Types](#)
for information about mapping to Dremio data types.

Was this page helpful?

[\[Previous](#)

[Oracle\]\(/current/data-sources/databases/oracle\)\[Next](#)

[SAP HANA\]\(/current/data-sources/databases/sap-hana\)](#)

[Configuring PostgreSQL as a Source](#)

[General](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Metadata](#)

[Privileges](#)

[Updating a PostgreSQL Source](#)

[Deleting a PostgreSQL Source](#)

[Predicate Pushdowns](#)

[Running Queries Directly on PostgreSQL Through Dremio](#)

-

For More Information

Source:
<https://docs.dremio.com/current/data-sources/databases/sap-hana>

Version: current [26.x]

On this page

SAP HANA Enterprise

Dremio supports connecting to SAP HANA directly via username and password. The connector was tested against HANA Express.

Requirements

To connect to SAP HANA, you need:

SAP HANA 2.0

SAP username and password

Dremio Configuration

On the Datasets page, to the right of **Sources** in the left panel, click !This is the Add Source icon.

In the Add Data Source dialog, under **Databases**, select the source.

The new source dialog box appears, which contains the following tabs:

General: Create a name for your database, specify the connection details, and set the authentication.

Advanced Options: (Optional) Set the advanced configuration options for your database.

Reflection Refresh: (Optional) Set a policy to control how often Reflections are refreshed and expired.

Metadata: (Optional) Specify dataset handling and metadata refresh.

Privileges: (Optional) Add privileges for users or roles.

Refer to the following sections for guidance on how to edit each tab.

General

To configure the source connection:

For **Name**, enter the name to identify the database in Dremio. The name cannot

include the following special characters: `/`, `:`, `[`, or `]`.

For **Host**, enter the hostname or IP address for the SAP HANA source.

For **Port**, enter the SAP HANA port number. The default port is `39017`.

For **Database**, enter the service name of your database.

For **Username**, enter the database username.

For **Password**, choose an authentication method:

No Authentication: Dremio does not attempt to provide any authentication when connecting with the SQL pool.

Master Credentials: Dremio must provide a specified username and password in order to access the SQL pool.

For **Username**, enter the database username.

For **Password**, choose a method:

Dremio: Provide the database password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for the Azure Key Vault secret that stores the Vertica password. The URI format is
`https://<vault_name>.vault.azure.net/secrets/<secret_name>` (for example,
`https://myvault.vault.azure.net/secrets/mysecret`).

note

To use Azure Key Vault as your application secret store, you must:

Deploy Dremio on Azure.

Complete the Requirements for Authenticating with Azure Key Vault.

It is not necessary to restart the Dremio coordinator when you rotate secrets stored in Azure Key Vault. Read Requirements for Secrets Rotation for more information.

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Choose the HashiCorp secrets engine you're using from the dropdown menu and enter the secret reference for the password in the correct format in the provided field.

note

Sources containing a large number of files or tables may take longer to be added. During this time, the source name is grayed out and shows a spinner icon, indicating the source is being added. Once complete, the source becomes accessible.

Advanced Options

Set the advanced configuration options for your database:

Record fetch size: Number of records to fetch at once. Set to `0` to have Dremio automatically decide. By default, this is set to `10`.

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to `8`.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to `60`.

Query timeout (s): The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Enable external authorization plugin: When enabled, authorizes an external plugin.

Connection Properties: Connection properties and values for the data source.

Reflection Refresh

Set the policy that controls how often Reflections are refreshed or expired, using the following options:

Never refresh: Select to prevent automatic Reflection refresh; otherwise, the default is to refresh automatically.

Refresh every: How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if **Never refresh** is selected.

Set refresh schedule: Specify the daily or weekly schedule.

Never expire: Select to prevent Reflections from expiring; otherwise, the default is to expire automatically after the time limit specified in **Expire after**.

Expire after: The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if **Never expire** is selected.

Metadata

Set the following metadata options:

Remove dataset definitions if underlying data is unavailable: Checked by default. If this box is *not* checked and the underlying files under a folder are removed or the folder/source is not accessible, Dremio does not remove the dataset definitions. This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Data Discovery: Set the time interval for fetching top-level source object names such as databases and tables. You can choose to set the **Fetch every** frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality. Use these parameters to fetch or expire the metadata:

Fetch mode: Fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to **Only Queried Datasets**.

Fetch every: Set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.

Expire after: Set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.

Privileges

To grant privileges to specific users or roles:

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

See [Access Control](#) for additional information about privileges.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

`%`, `*`, `+`, `-`, `/`
`<`, `<=`, `<>`, `=`, `>`, `>=`, `!=`
AND, NOT, OR, `||`
ABS
ACOS
ADD_MONTHS
ASIN
ATAN
ATAN2
AVG
CAST
CBRT
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COT
DATE_ADD
DATE_DIFF
DATE_SUB
DATE_TRUNC_CENTURY

DATE_TRUNC_DAY
DATE_TRUNC_DECADE
DATE_TRUNC_HOUR
DATE_TRUNC_MILLENIUM
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_SECOND
DATE_TRUNC_YEAR
DEGREES
E
EXP
EXTRACT_DAY
EXTRACT_DOW
EXTRACT_DOY
EXTRACT_HOUR
EXTRACT_MINUTE
EXTRACT_MONTH
EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LCASE
LEFT
LN
LOCATE
LOG
LOG10
LPAD
LTRIM
MAX
MIN
MOD
MONTH
PI
POSITION
POW
POWER
RADIANS
RAND
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
SUBSTR

SUBSTRING
SUM
TAN
TO_CHAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
YEAR

Data Source Management

Updating the Source

To update the source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the dropdown.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name.

Click **Save**.

Deleting the Source

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the `ADMIN` role can delete the source.

To delete the source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and click !The Settings icon to the right.

From the dropdown, select **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Sources containing a large number of files or tables may take longer to be removed. During this time, the source name is grayed out and shows a spinner icon, indicating the source is being removed. Once complete, the source disappears.

Querying the SAP HANA Source Directly

Dremio users can run pass queries through Dremio to run on your database. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[\[Previous](#)

[PostgreSQL\]\(/current/data-sources/databases/postgres\)\[Next](#)

[Snowflake\]\(/current/data-sources/databases/snowflake\)](#)

[Requirements](#)

[Dremio Configuration](#)

[General](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Metadata](#)

[Privileges](#)

[Predicate Pushdowns](#)

[Data Source Management](#)

[Updating the Source](#)

[Deleting the Source](#)

[Querying the SAP HANA Source Directly](#)

Source:
<https://docs.dremio.com/current/data-sources/databases/snowflake>

Version: current [26.x]

On this page

Snowflake

[Snowflake](#) is a cloud data warehouse.

Prerequisite

Ensure that your Dremio cluster is at version 23.1 or later.

User Impersonation

Dremio supports OAuth with impersonation for Snowflake. This allows Dremio users to authenticate via external OAuth and map to Snowflake roles securely. For reference, see [Snowflake's Create Security Integration \(External OAuth\) documentation](#).

note

Reflections are not supported on data sources with user impersonation enabled to ensure that all security and governance policies defined in the underlying data source are enforced.

Reflections created prior to enabling user impersonation must be manually dropped, as they will fail to refresh once impersonation is active.

Before configuring a Snowflake source with user impersonation, perform the following steps:

Run the following curl commands to obtain the Dremio OAuth parameters (issuer and public key):

To get the issuer:

```
curl --location 'https://<dremio_url>/api/v3/external-oauth/discovery/jwt-issuer' \
--header 'Authorization: Bearer <Token>' \
--header 'Content-Type: application/json' \
--data ''
```

To get the public key:

```
curl --location 'https://<dremio_url>/api/v3/external-oauth/discovery/jwks' \
--header 'Authorization: Bearer <Token>' \
--header 'Content-Type: application/json' \
--data ''
```

The above JWKS response needs to be converted to PEM format, which Snowflake accepts. We recommend using this open-source tool: [rsa-jwks-to-pem](#).

Example conversion:

```
python rsa-jwks-to-pem.py key_jwks.json
```

Create a [Snowflake external OAuth security integration](#) in Snowflake.

Set `EXTERNAL_OAUTH_ISSUER` to the issuer obtained from Dremio, `EXTERNAL_OAUTH_RSA_PUBLIC_KEY` to the PEM-formatted key from the script, and `EXTERNAL_OAUTH_AUDIENCE_LIST` to any additional audience values for token validation beyond your Snowflake account URL.

Create Security Integration

```
CREATE OR REPLACE SECURITY INTEGRATION snowflake_imp
TYPE = EXTERNAL_OAUTH
ENABLED = TRUE
EXTERNAL_OAUTH_TYPE = CUSTOM
```

```
EXTERNAL_OAUTH_ISSUER = '<issuer-from-dremio>'
EXTERNAL_OAUTH_AUDIENCE_LIST = ('<audience-values>')
EXTERNAL_OAUTH_ALLOWED_ROLES_LIST = ('REGRESSION', 'ACCOUNTADMIN', 'PUBLIC')
EXTERNAL_OAUTH_RSA_PUBLIC_KEY = '<PEM-formatted-key>'
EXTERNAL_OAUTH_TOKEN_USER_MAPPING_CLAIM = 'sub'
EXTERNAL_OAUTH_SNOWFLAKE_USER_MAPPING_ATTRIBUTE = 'login_name';
```

To configure Snowflake source in any mode (which allows users to assume any role they have access to in Snowflake), enable `EXTERNAL_OAUTH_ANY_ROLE_MODE` for Snowflake security integration:

Alter Security Integration

```
ALTER SECURITY INTEGRATION snowflake_imp SET EXTERNAL_OAUTH_ANY_ROLE_MODE = 'ENABLE';
```

Configuring Snowflake as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Snowflake**.

General

In the **Name** field, specify the name by which you want the Snowflake source to appear in the **Databases** section. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Under **Connection**, follow these steps:

note

The optional connection parameters are case-sensitive. For example, if the name of a warehouse uses upper case only (e.g., WAREHOUSE1), specify it the same way in the **Warehouse** field.

In the **Host** field, specify the hostname of the Snowflake source in this format: `LOCATOR_ID.snowflakecomputing.com`.

(Optional) In the **Database** field, specify the default database to use.

(Optional) In the **Role** field, specify the default access-control role to use.

(Optional) In the **Schema** field, specify the default schema to use.

(Optional) In the **Warehouse** field, specify the warehouse that will provide resources for executing DML statements and queries.

Under **Authentication**, select either Login-password authentication, Key-pair authentication or OAuth with impersonation:

Login-password authentication: In the **Username** field, specify the Snowflake username. Under **Password**, in the dropdown menu, choose a method for providing the

Snowflake password. If you choose Dremio, provide the Snowflake password in plain text in the provided field. Dremio stores the password. You may also choose to use one of the supported secrets managers to provide the Snowflake password:

Azure Key Vault: Provide the URI for the Azure Key Vault secret that stores the Snowflake password. The URI format is ``https://<vault_name>.vault.azure.net/secrets/<secret_name>`` (for example, ``https://myvault.vault.azure.net/secrets/mysecret``).

note

To use Azure Key Vault as your application secret store, you must:

Deploy Dremio on Azure AKS.

Complete the Requirements for Authenticating with Azure Key Vault.

It is not necessary to restart the Dremio coordinator when you rotate secrets stored in Azure Key Vault. Read Requirements for Secrets Rotation for more information.

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the Snowflake password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Choose the HashiCorp secrets engine you're using from the dropdown menu and enter the secret reference for the Snowflake password in the correct format in the provided field.

Key-pair authentication (see Snowflake's key-pair documentation): In the **Username** field, specify the Snowflake username. Under **Private Key** and **Private key passphrase**, in the dropdown menus, choose a method for providing the Snowflake private key and private key passphrase, respectively. If you choose Dremio, provide the Snowflake private key and private key passphrase in plain text in the provided fields. Dremio stores the private key and private key passphrase. You may also choose to use one of the supported secrets managers to provide the Snowflake private key and private key passphrase:

Azure Key Vault: Provide the URI for the Azure Key Vault secret that stores the Snowflake private key and private key passphrase. The URI format is ``https://<vault_name>.vault.azure.net/secrets/<secret_name>`` (for example, ``https://myvault.vault.azure.net/secrets/mysecret``).

note

To use Azure Key Vault as your application secret store, you must:

Deploy Dremio on Azure AKS.

Complete the Requirements for Authenticating with Azure Key Vault.

It is not necessary to restart the Dremio coordinator when you rotate secrets stored in Azure Key Vault. Read Requirements for Secrets Rotation for more information.

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the Snowflake private key and private key passphrase, which is available in the AWS web console or using command line tools.

-

HashiCorp Vault: Choose the HashiCorp secrets engine you're using from the dropdown menu and enter the secret reference for the Snowflake private key and private key passphrase in the correct format in the provided field.

OAuth with impersonation: This allows Dremio users to authenticate via external OAuth and map to Snowflake roles securely. If you have not already, complete the steps in User Impersonation.

Set the JWT `audience` parameter to match Snowflake's `EXTERNAL_OAUTH_AUDIENCE_LIST`. This ensures proper token validation and role mapping between Dremio and Snowflake.

Advanced Options

On the Advanced Options page, you can set values for these non-required options:

Option	Description
---	---
Maximum Idle Connections	The total number of connections allowed to be idle at a given time. The default maximum idle connections is 8.
Connection Idle Time	The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. The default connection idle time is 60 seconds.
Query Timeout	The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.
Record Fetch Size	The maximum number of records to allow a single query to fetch. This setting prevents queries from using too many resources.

Reflection Refresh

On the Reflection Refresh page, set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata

On the Metadata page, you can configure settings to refresh metadata and handle datasets.

Dataset Handling

These are the optional **Dataset Handling** parameters.

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

---	---
-----	-----

Remove dataset definitions if underlying data is unavailable	By default, Dremio removes dataset definitions if underlying data is unavailable. Useful when files are temporarily deleted and added back in the same location with new sets of files.
---	---

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

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

---	---
-----	-----

(Optional) Fetch every	You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour.
-------------------------------	---

Dataset Details: The metadata that Dremio needs for query planning such as information required for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

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

---	---
-----	-----

Fetch mode	You can choose to fetch only from queried datasets that are set by default. Dremio updates details for previously queried objects in a source. Fetching from all datasets is deprecated.
-------------------	--

Fetch every	You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour.
--------------------	---

Expire after	You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours.
---------------------	--

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Snowflake Source

To update a Snowflake source:

-

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [!The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Snowflake as a Source](#).

Click **Save**.

Deleting a Snowflake Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Snowflake source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Upgrading from Dremio Hub's Community Snowflake Plugin

caution

Removing a Snowflake source will drop all tables in the source. If you have any Reflections configured on tables or table-level ACLs (customized privileges) in your Snowflake sources, copy the details of those items before you remove any sources. After upgrading and re-adding your sources, you will need to recreate those Reflections and ACLs.

Views are not affected by removing and re-adding Snowflake sources, provided the sources are re-added with the same names.

note

The community Snowflake plugin from Dremio Hub is not compatible with Dremio

version 23.0 and later.

You should use Dremio version 23.1 or later if you have Snowflake sources because it comes with an official Snowflake plugin.

If you are upgrading an older version of Dremio to version 23.1 or later, you must do the following:

Note the details of any Reflections and ACLs configured on tables in Snowflake sources.

Remove your Snowflake sources from Dremio.

Remove the community Snowflake plugin and the existing Snowflake JDBC driver.

Upgrade Dremio to version 23.1 or later.

Add your Snowflake sources to Dremio with the same names.

Recreate any table-level Reflections and ACLs on your Snowflake sources.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

`||`, AND, OR
`+`, `-`, `/`, `*`
`<=`, `<`, `>`, `>=`, `=`, `<>`, `!=`

ABS
ADD_MONTHS
AVG
BETWEEN
CASE
CAST
CEIL
CEILING
CHARACTER_LENGTH
CHAR_LENGTH
COALESCE
CONCAT
COUNT
COUNT_DISTINCT
COUNT_DISTINCT_MULTI
COUNT_FUNCTIONS
COUNT_MULTI
COUNT_STAR
DATE_ADD
DATE_SUB
DATE_TRUNC
DATE_TRUNC_DAY
DATE_TRUNC_HOUR
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER

DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DAYOFMONTH
DAYOFWEEK
DAYOFYEAR
EXTRACT
FLOOR
ILIKE
IN
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST_DAY
LEFT
LENGTH
LIKE
LOCATE
LOWER
LPAD
LTRIM
MAX
MEDIAN
MIN
MOD
NOT
PERCENT_CONT
PERCENT_DISC
PERCENT_RANK
POSITION
REGEXP_LIKE
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SQRT
STDDEV
STDDEV_POP
STDDEV_SAMP
SUBSTR
SUBSTRING
SUM
TO_CHAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UPPER

Running Queries Directly on Snowflake Through
Dremio

Dremio users can run pass queries through Dremio to run on Snowflake. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[\[Previous](#)

[SAP HANA\]\(/current/data-sources/databases/sap-hana\)\[Next](#)

[Teradata\]\(/current/data-sources/databases/teradata\)](#)

[Prerequisite](#)

[User Impersonation](#)

[Configuring Snowflake as a Source](#)

[General](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Metadata](#)

[Privileges](#)

[Updating a Snowflake Source](#)

[Deleting a Snowflake Source](#)

[Upgrading from Dremio Hub's Community Snowflake Plugin](#)

[Predicate Pushdowns](#)

[Running Queries Directly on Snowflake Through Dremio](#)

Source:
<https://docs.dremio.com/current/data-sources/databases/teradata>

Version: current [26.x]

On this page

Teradata Enterprise

This topic describes Teradata data source setup and Dremio configuration.

User Impersonation

The Teradata database username provided in the source configuration is the default username that is used for running queries. When queries are run against Teradata in Dremio, users use the privileges associated with the Teradata database username and

run queries under that username.

You can change this default in Dremio by enabling user impersonation in the Advanced Options, which allows users to run queries under their own usernames and restricts their access. For example, `user_1` can run queries as `user_1` rather than `Teradata_svc`. Before enabling user impersonation, some setup is required in Teradata to allow one user to impersonate another user, because the username of the user in Dremio must be the same as their username in Teradata and the user must be able to connect through the Teradata database username.

To set up user impersonation, follow these steps:

Ensure the user's username in Teradata matches their username in Dremio. If the usernames do not match, modify one of the usernames or create a new user account with a matching username.

Run a GRANT CONNECT THROUGH command in Teradata to allow the user to connect through the Teradata database username:

Example of granting the CONNECT THROUGH privilege in Teradata

```
GRANT CONNECT THROUGH proxyuser TO PERMANENT testuser1 WITHOUT ROLE;
```

In this example, the user can log in as `testuser1` in Dremio and in Teradata, and they can connect through the `proxyuser`. The `proxyuser` is the Teradata database username provided in the source configuration.

Log in as an admin to Dremio.

Follow the steps for Dremio Configuration using the Teradata database username and enable **User Impersonation** in the **Advanced Options**.

Grant source privileges to the user.

Now that you have enabled user impersonation, a user logging in to Dremio with their username can access the Teradata source and its datasets according to their privileges. The user also runs queries against Teradata under their username.

Teradata Setup

Dremio provides the Teradata connector with Dremio Enterprise Edition. You must install the proprietary Teradata JDBC driver in order to connect to a Teradata source.

To setup Teradata as a data source:

Download the Teradata JDBC jars: **tdgssconfig.jar** and **TeraJDBC.jar**. The Teradata JDBC driver version 16.20+ does not need the **tdgssconfig.jar** file.

Move the jar files into the **/opt/dremio/jars/3rdparty** directory on every Dremio node.

Restart Dremio coordinators and executors to pick up the newly-installed JDBC driver.

Download Teradata Dremio plugin JAR and move it into the **/opt/dremio/jars** directory.

Configuring Teradata as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Teradata**.

General

Under **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Connection

Name	Description
---	---
Host	Teradata host name.
Port	Teradata port number. Defaults to 1025.
Service Name	Service Name of your database.

Authentication

Select an authentication option:

Username: Teradata username

Password: Select the password store from the dropdown menu:

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Select the **Encrypt connection** option to encrypt the connection to Teradata. Clear the checkbox to disable encryption.

Advanced Options

Specify advanced options with the following settings.

Show only the initial database used for connecting.

Record fetch size: Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. Default: 10

Maximum idle connections: The total number of connections allowed to be idle at a

given time. By default, this is set to *8*.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to *60*.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

User Impersonation: Allows users to run queries using their credentials rather than the username provided in the source credentials. Some setup is required in Teradata to allow one user to impersonate another user. See User Impersonation.

Reflection Refresh

Specify refresh policy options with the following settings.

Never refresh Specifies how often to refresh based on hours, days, weeks, or never.

Never expire Specifies how often to expire based on hours, days, weeks, or never.

Metadata

Specify metadata options with the following settings.

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is *not* checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

Dataset Discovery Refresh interval for top-level source object names such as names of DBs and tables.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Dataset Details The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality.

Fetch mode Specify either Only Queried Datasets, All Datasets, or As Needed. Default: Only Queried Datasets

Only Queried Datasets Dremio updates details for previously queried objects in a source.

This mode increases query performance because less work is needed at query time for these datasets.

All Datasets Dremio updates details for all datasets in a source.
This mode increases query performance because less work is needed at query time.

As Needed Dremio updates details for a dataset at query time.
This mode minimized metadata queries on a source when not used,
but might lead to longer planning times.

Fetch every Specify fetch time based on minutes, hours, days, or weeks. Default: 1 hour

Expire after Specify expiration time based on minutes, hours, days, or weeks. Default: 3 hours

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges.

note

All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Teradata Source

To update a Teradata source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Teradata as a Source](#).

Click **Save**.

Deleting a Teradata Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can

delete the source.

To delete a Teradata source, perform these steps:

On the Datasets page, click **Sources** > **Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

`<`,`<=`,`<>`,`=`,`>`,`>`,`!=`

`*`,`+`,`-`,`/`

AND, NOT, OR, `||`

ABS

ACOS

ADD_MONTHS

ASIN

ATAN

ATAN2

AVG

CAST

CEIL

CEILING

CHAR_LENGTH

CHARACTER_LENGTH

CONCAT

COS

COSH

COT

COVAR_POP

COVAR_SAMP

DATE_ADD

DATE_SUB

DATE_TRUNC_DAY

DATE_TRUNC_MONTH

DATE_TRUNC_QUARTER

DATE_TRUNC_WEEK

DATE_TRUNC_YEAR

DEGREES

E

EXP
EXTRACT\ _DAY
EXTRACT\ _DOW
EXTRACT\ _DOY
EXTRACT\ _HOUR
EXTRACT\ _MINUTE
EXTRACT\ _MONTH
EXTRACT\ _QUARTER
EXTRACT\ _SECOND
EXTRACT\ _WEEK
EXTRACT\ _YEAR
FLOOR
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
LAST\ _DAY
LCASE
LEFT
LENGTH
LIKE
LN
LOCATE
LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MIN
MOD
PI
POSITION
POW
POWER
RADIANS
REPLACE
REVERSE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SINH
SQRT
STDDEV
STDDEV\ _POP
STDDEV\ _SAMP
SUBSTR
SUBSTRING
SUM
TAN
TANH

TIMESTAMPADD_DAY
TIMESTAMPADD_HOUR
TIMESTAMPADD_MINUTE
TIMESTAMPADD_MONTH
TIMESTAMPADD_SECOND
TIMESTAMPADD_YEAR
TIMESTAMPDIFF_YEAR
TO_CHAR
TO_DATE
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
VAR_POP
VAR_SAMP

Running Queries Directly on Teradata Through Dremio

Dremio users can run pass queries through Dremio to run on Teradata. Doing so can sometimes decrease query execution times. For more information, see [Querying Relational-Database Sources Directly](#).

Was this page helpful?

[\[Previous](#)

[Snowflake\]\(/current/data-sources/databases/snowflake\)\[Next](#)

[Vertica\]\(/current/data-sources/databases/vertica\)](#)

[User Impersonation](#)

[Teradata Setup](#)

[Configuring Teradata as a Source](#)

[General](#)

[Advanced Options](#)

[Reflection Refresh](#)

[Metadata](#)

[Privileges](#)

[Updating a Teradata Source](#)

[Deleting a Teradata Source](#)

[Predicate Pushdowns](#)

[Running Queries Directly on Teradata Through Dremio](#)

Source:
<https://docs.dremio.com/current/data-sources/databases/vertica>

Version: current [26.x]

On this page

Vertica

Vertica is an analytical database.

Prerequisites

Ensure that you have the following details before configuring Vertica as a source:

Database name

Hostname or IP address

Port

Configuring Vertica as a Source

On the Datasets page, to the right of **Sources** in the left panel, click [!This is the Add Source icon.](#)

In the Add Data Source dialog, under **Databases**, select **Vertica**.

General

For **Name**, enter the name to identify the data source in Dremio. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Connection

Name Description
--- ---
Host Vertica host name.
Port Vertica port number. Defaults to 5433.
Database Service name of your database.

Authentication

Select an authentication option:

Username: Vertica username

Password: Select the password store from the dropdown menu:

-

Dremio: Provide the password in plain text. Dremio stores the password.

Azure Key Vault: Provide the URI for your stored password using the format ``https://<vault_name>.vault.azure.net/secrets/<secret_name>``

AWS Secrets Manager: Provide the Amazon Resource Name (ARN) for the AWS Secrets Manager secret that holds the password, which is available in the AWS web console or using command line tools.

HashiCorp Vault: Select your HashiCorp secrets engine from the dropdown and enter the password reference in the required format.

Advanced Options

Specify advanced options with the following settings.

Record fetch size: Number of records to fetch at once. Set to 0 (zero) to have Dremio automatically decide. By default, this is set to **10**.

Maximum idle connections: The total number of connections allowed to be idle at a given time. By default, this is set to **8**.

Connection idle time (s): The amount of time (in seconds) allowed for a connection to remain idle before the connection is terminated. By default, this is set to **60**.

Query timeout: The amount of time (in seconds) allowed to wait for the results of a query. If this time expires, the connection being used is returned to an idle state.

Connection Properties: Connection properties and values for the data source.

Reflection Refresh

You can set the policy that controls how often Reflections are scheduled to be refreshed automatically, as well as the time limit after which Reflections expire and are removed.

Option	Description
---	---
Never refresh	Select to prevent automatic Reflection refresh, default is to automatically refresh.
Refresh every	How often to refresh Reflections, specified in hours, days or weeks. This option is ignored if Never refresh is selected.
Set refresh schedule	Specify the daily or weekly schedule.
Never expire	Select to prevent Reflections from expiring, default is to automatically expire after the time limit below.
Expire after	The time limit after which Reflections expire and are removed from Dremio, specified in hours, days or weeks. This option is ignored if Never expire is selected.

Metadata

Specifying metadata options is handled with the following settings.

Dataset Handling

Remove dataset definitions if underlying data is unavailable (Default).

If this box is **not** checked and the underlying files under a folder are removed or the folder/source is not accessible,

Dremio does not remove the dataset definitions.

This option is useful in cases when files are temporarily deleted and put back in place with new sets of files.

Metadata Refresh

These are the optional **Metadata Refresh** parameters:

Dataset Discovery: The refresh interval for fetching top-level source object names such as databases and tables. Set the time interval using this parameter.

| Parameter | Description |

| --- | --- |

| **Fetch every** | You can choose to set the frequency to fetch object names in minutes, hours, days, or weeks. The default frequency to fetch object names is 1 hour. |

Dataset Details: The metadata that Dremio needs for query planning such as information needed for fields, types, shards, statistics, and locality. These are the parameters to fetch the dataset information.

| Parameter | Description |

| --- | --- |

| **Fetch mode** | You can choose to fetch only from queried datasets. Dremio updates details for previously queried objects in a source. By default, this is set to **Only Queried Datasets**. |

| **Fetch every** | You can choose to set the frequency to fetch dataset details in minutes, hours, days, or weeks. The default frequency to fetch dataset details is 1 hour. |

| **Expire after** | You can choose to set the expiry time of dataset details in minutes, hours, days, or weeks. The default expiry time of dataset details is 3 hours. |

Privileges

On the Privileges tab, you can grant privileges to specific users or roles. See [Access Controls](#) for additional information about privileges. All privileges are optional.

For **Privileges**, enter the user name or role name that you want to grant access to and click the **Add to Privileges** button. The added user or role is displayed in the **USERS/ROLES** table.

For the users or roles in the **USERS/ROLES** table, toggle the checkmark for each privilege you want to grant on the Dremio source that is being created.

Click **Save** after setting the configuration.

Updating a Vertica Source

To update a Vertica source:

On the Datasets page, under **Databases** in the panel on the left, find the name of the source you want to update.

Right-click the source name and select **Settings** from the list of actions. Alternatively, click the source name and then the [The Settings icon](#) at the top right corner of the page.

In the **Source Settings** dialog, edit the settings you wish to update. Dremio does not support updating the source name. For information about the settings options, see [Configuring Vertica as a Source](#).

Click **Save**.

Deleting a Vertica Source

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

To delete a Vertica source, perform these steps:

On the Datasets page, click **Sources > Databases** in the panel on the left.

In the list of data sources, hover over the name of the source you want to remove and right-click.

From the list of actions, click **Delete**.

In the Delete Source dialog, click **Delete** to confirm that you want to remove the source.

note

Deleting a source causes all downstream views that depend on objects in the source to break.

Predicate Pushdowns

Dremio delegates the execution of these expressions and functions to the database being queried, often dramatically improving query performance. It can also offload entire SQL queries that include one or more of these expressions and functions.

``*`,`+`,`-`,`/`,`%`,`<`,`<=`,`<>`,`=`,`>`,`>=`,`!=`,`AND`,`NOT`,`OR`,`||``
ABS
ACOS
ADD_MONTHS
ASIN
ATAN
ATAN2
AVG

BTRIM
CAST
CBRT
CEIL
CEILING
CHAR_LENGTH
CHARACTER_LENGTH
CONCAT
COS
COT
DATE_ADD
DATE_PART
DATE_SUB
DATE_TRUNC_CENTURY
DATE_TRUNC_DAY
DATE_TRUNC_DECADE
DATE_TRUNC_HOUR
DATE_TRUNC_MICROSECOND
DATE_TRUNC_MILLISECOND
DATE_TRUNC_MINUTE
DATE_TRUNC_MONTH
DATE_TRUNC_QUARTER
DATE_TRUNC_SECOND
DATE_TRUNC_WEEK
DATE_TRUNC_YEAR
DEGREES
E
EXP
EXTRACT_CENTURY
EXTRACT_DAY
EXTRACT_DECADE
EXTRACT_DOW
EXTRACT_DOY
EXTRACT_EPOCH
EXTRACT_HOUR
EXTRACT_MILLENNIUM
EXTRACT_MINUTE
EXTRACT_MONTH
EXTRACT_QUARTER
EXTRACT_SECOND
EXTRACT_WEEK
EXTRACT_YEAR
FLOOR
ILIKE
IS DISTINCT FROM
IS NOT DISTINCT FROM
IS NOT NULL
IS NULL
ISNULL
LAST_DAY
LCASE
LEFT
LENGTH
LIKE

LN
LOCALTIME
LOCALTIMESTAMP
LOCATE
LOG
LOG10
LOWER
LPAD
LTRIM
MAX
MIN
MOD
NOW
NULLIF
PI
POSITION
POW
POWER
RADIANS
RANDOM
REGEXP_REPLACE
REPLACE
RIGHT
ROUND
RPAD
RTRIM
SIGN
SIN
SQRT
STDDEV
STDDEV_POP
STDDEV_SAMP
STRPOS
SUBSTR
SUBSTRING
SUM
TAN
TIMESTAMPADD
TIMESTAMPDIFF
TO_CHAR
TO_DATE
TO_NUMBER
TO_TIMESTAMP
TRIM
TRUNC
TRUNCATE
UCASE
UPPER
VAR_POP
VAR_SAMP
VARIANCE

Was this page helpful?

[Previous

Teradata](/current/data-sources/databases/teradata)

Prerequisites

Configuring Vertica as a Source

General

Advanced Options

Reflection Refresh

Metadata

Privileges

Updating a Vertica Source

Deleting a Vertica Source

Predicate Pushdowns

Source: dremio-software-deploy-dremio.md

Dremio Software - Deploy Dremio

Source:
<https://docs.dremio.com/current/deploy-dremio/>

Version: current [26.x]

On this page

Deploy Dremio

This topic describes the deployment models. Dremio is a distributed system that can be deployed in a public cloud or on-premises. A Dremio cluster can be co-located with one of the data sources (Hadoop or NoSQL database) or deployed separately.

Deploy on Kubernetes

Kubernetes is the recommended deployment option for Dremio. For more information, see the following topics in this section:

[Kubernetes Environments](#) - Learn about the Kubernetes environments used to deploy Dremio.

[Deploying on Kubernetes](#) - Deploy Dremio on your Kubernetes environment.

[Configuring Your Values](#) - Understand the configuration of your deployments in more detail.

Managing Engines – Manage Dremio engines to optimize query execution.

Other Deployment Options

Besides Kubernetes, Dremio provides other options for deployment described in this section.

Shared Multi-Tenant Environment

If you plan on using a shared multi-tenant environment, Dremio provides a model that uses YARN for deployment:

****Hadoop using YARN**** - Dremio on Hadoop in YARN deployment. Dremio integrates with YARN ResourceManager to secure compute resources in a shared multi-tenant environment.

note

Co-locating Dremio with Hadoop/NoSQL: When Dremio is co-located with a Hadoop cluster (such as HDFS) or distributed NoSQL database (such as Elasticsearch or MongoDB), it is important to utilize containers (cgroups, Docker, and YARN containers) to ensure adequate resources for each process.

Dremio features a high-performance asynchronous engine that minimizes the number of threads and context switches under heavy load. So, unless containers are utilized, the operating system may over-allocate resources to other thread-hungry processes on the nodes.

Standalone Cluster

If you plan on creating a standalone cluster, Dremio provides the flexibility to deploy Dremio as a standalone on-premise cluster:

****Standalone Cluster**** - Dremio on a standalone on-premise cluster. In this scenario, a Hadoop cluster is not available and the data is not in a single distributed NoSQL database.

Was this page helpful?

[\[Previous](#)

[Architecture\]\(/current/what-is-dremio/architecture\)\[Next](#)

[Kubernetes Environments\]\(/current/deploy-dremio/kubernetes-environments\)](#)

[Deploy on Kubernetes](#)

[Other Deployment Options](#)

[Shared Multi-Tenant Environment](#)

[Standalone Cluster](#)

Source:
<https://docs.dremio.com/current/deploy-dremio/kubernetes-environments>

Version: current [26.x]

On this page

Kubernetes Environments for Dremio

Dremio is designed to run Kubernetes environments, providing enterprise-grade data lakehouse capabilities. To successfully [deploy Dremio on Kubernetes](#), you need a compatible hosted Kubernetes environment.

Dremio is tested and supported on the following Kubernetes environments:

Elastic Kubernetes Service (EKS)

Azure Kubernetes Service (AKS)

Google Kubernetes Engine (GKE)

Red Hat OpenShift

The sections on this page detail recommendations for AWS and Azure. Please use the information provided as a guide for your vendors' equivalent options.

note

If you're using a containerization platform built on Kubernetes that isn't listed here, please contact your provider and Dremio Account team to discuss compatibility and support options.

Requirements

Versions

Dremio requires regular updates to your Kubernetes version. You must be on an officially supported version, and preferably not one on extended support. See the following examples for AWS [Available versions on standard support](#) and Azure [Kubernetes versions](#).

Recommendations

See this table for resource request recommendations of the various parts of the deployment, [Recommended Resources Configuration](#).

For a list of all Dremio engine sizes see, [Add an Engine](#). Engines will make up the lions share of any Dremio deployment.

Node Sizes

The following sections suggest AWS and Azure machines that could be used to meet our recommendations.

Dremio recommends having separate EKS node groups for the different components of our services to allow each node group to autoscale independently:

Core Services

Coordinators

For coordinators, Dremio recommends at least 32 CPUs and 64 GB of memory, hence, a ``c6i.8xlarge`` or ``Standard_F32s_v2`` is a good option, offering a CPU-to-memory ratio of 1:2. In the Helm charts, this would result in 30 CPUs and 60 GB of memory allocated to the Dremio pod.

Executors

For executors, Dremio recommends either:

16 CPUs and 128 GB of memory, hence, a ``r5d.4xlarge`` or ``Standard_E16_v5`` is a good option, offering a CPU-to-memory ratio of 1:8. In the Helm charts, this results in 15 CPUs and 120 GB of memory allocated to the Dremio pod.

32 CPUs and 128 GB of memory, hence, a ``m5d.8xlarge`` or ``Standard_D32_v5`` is a good option, offering a CPU-to-memory ratio of 1:4 for high-concurrency workloads. In the Helm charts, this results in 30 CPUs and 120 GB of memory allocated to the Dremio pod.

Auxiliary Services

Open Catalog and Semantic Search.

Catalog is made up of 4 key components: Catalog Service, Catalog Server, Catalog External, and MongoDB. Search has one key component, OpenSearch.

Each of these components needs between 2-4 CPUs and 4-16 GB of memory; hence, a ``m5d.2xlarge`` or ``Standard_D8_v5`` is a good option and could be used to host multiple containers that are part of these services.

ZooKeeper, NATS, Operators, and Open Telemetry:

Each of these need between 0.5-1 CPUs and 0.5-1 GB, ``m5d.large``, ``t2.medium``, ``Standard_D2_v5`` or ``Standard_A2_v2`` are good options and could be used to host multiple containers that are part of these services.

Disk Storage Class

Dremio recommends:

For AWS, GP3 or IO2 as the storage type for all nodes.

For Azure managed-premium as the storage type for all nodes.

Additionally, for coordinators and executors, you can further use local NVMe SSD storage for C3 and spill on executors. For more information on storage classes, see the following resources [AWS Storage Class](#) and [Azure Storage Class](#).

Storage size requirements are:

Coordinator volume #1: 128-512 GB (key-value store).

Coordinator volume #2: 16 GB (logs).

Executor volume #1: 128-512 GB (spilling).

Executor volume #2: 128-512 GB (C3).

Executor volume #3: 16 GB (logs).

MongoDB volume: 128-512 GB.

OpenSearch volume: 128 GB.

Zookeeper volume: 16 GB.

EKS Add-Ons

The following add-ons are required for EKS clusters:

Amazon EBS CSI Driver

EKS Pod Identity Agent

Was this page helpful?

[Previous

Deploy Dremio](/current/deploy-dremio/)[Next

Deploy on Kubernetes](/current/deploy-dremio/deploy-on-kubernetes)

Requirements

Versions

Recommendations

EKS Add-Ons

Source:
<https://docs.dremio.com/current/deploy-dremio/deploy-on-kubernetes>

Version: current [26.x]

On this page

Deploy Dremio on Kubernetes

You can follow these instructions to deploy Dremio on Kubernetes provisioned through a cloud provider or running in an on-premises environment.

FREE TRIAL

If you are using an **Enterprise Edition free trial**, go to [Get Started with the Enterprise Edition Free Trial](#).

Prerequisites

Before deploying Dremio on Kubernetes, ensure you have the following:

A hosted Kubernetes environment to deploy and manage the Dremio cluster. Each Dremio release is tested against [Amazon Elastic Kubernetes Service \(EKS\)](#), [Azure Kubernetes Service \(AKS\)](#), and [Google Kubernetes Engines \(GKE\)](#) to ensure compatibility. If you have a containerization platform built on top of Kubernetes that is not listed here, please contact your provider and the Dremio Account Team regarding compatibility.

Helm 3 installed on your local machine to run Helm commands. For installation instructions, refer to [Installing Helm](#) in the Helm documentation.

A local kubectl configured to access your Kubernetes cluster. For installation instructions, refer to [kubectl](#) in the Kubernetes documentation.

Object Storage: Amazon S3 (including S3-compatible, e.g., MinIO), Azure Storage, or Google Cloud Storage (GCS).

Storage classes that support ReadWriteOnce (RWO) access mode and ideally can create expandable volumes.

The ability to connect to [Quay.io](#) to access the [new v3 Helm chart](#) for Dremio 26+, since the [older v2 Helm chart](#) will not function.

Additional Prerequisites for the Enterprise Edition

For the Enterprise Edition, you must:

Create an account on [Quay.io](#) to access [Dremio's OCI repository](#), which stores Dremio's Helm charts and images.

To get access, contact your Dremio account executive or Dremio Support.

note

If your internet access doesn't allow reaching Dremio's OCI repository in Quay.io, consider using a private mirror to fetch Dremio's Helm chart images.

Get a valid license key issued by Dremio to put in the Helm chart. To obtain the license, refer to [Licensing](#).

Additional Prerequisites for the OpenShift

Before deploying Dremio onto OpenShift, you additionally need the following:

Have the OpenShift ``oc`` CLI command configured and authenticated. For the installation instructions, see [OpenShift CLI \(oc\)](#).

Node Tuning for OpenSearch on OpenShift

OpenSearch requires the `vm.max_map_count` kernel parameter to be set to at least **262144**.

This parameter controls the maximum number of memory map areas a process can have, and OpenSearch uses memory-mapped files extensively for performance.

Without this setting, OpenSearch pods will fail to start with errors related to virtual memory limits.

Since the Helm chart sets `setVMMaxMapCount: false` for OpenShift compatibility (to avoid privileged init containers), you need to configure this kernel parameter at the node level. The **recommended way** to do it is a Node Tuning Operator. This Operator ships with OpenShift and provides a declarative way to configure kernel parameters.

Create a `Tuned` resource to configure the required kernel parameter:

The `tuned-opensearch.yaml` configuration file

```
apiVersion: tuned.openshift.io/v1
kind: Tuned
metadata:
  name: openshift-opensearch
  namespace: openshift-cluster-node-tuning-operator
spec:
  profile:
    - data: |
        [main]
        summary=Optimize systems running OpenSearch on OpenShift nodes
        include=openshift-node
        [sysctl]
        vm.max_map_count=262144
        name: openshift-opensearch
  recommend:
    - match:
        - label: tuned.openshift.io/opensearch
          type: pod
        priority: 20
        profile: openshift-opensearch
```

This YAML should be saved locally and applied to any cluster you intend to deploy Dremio:

```
oc apply -f tuned-opensearch.yaml
```

Step 1: Deploy Dremio

To deploy the Dremio cluster in Kubernetes, do the following:

Configure your values to deploy Dremio to Kubernetes in the file `values-overrides.yaml`. For that, go to [Configuring Your Values to Deploy Dremio to Kubernetes](#) and get back here to continue with the deployment.

On your terminal, start the deployment by installing Dremio's Helm chart:

Standard Kubernetes

OpenShift

Run the following command for any Kubernetes environment except for OpenShift:

```
helm install <your-dremio-install-release> oci://quay.io/dremio/dremio-helm \
--values <your-local-path>/values-overrides.yaml \
--version <optional-helm-chart-version> \
--set-file <optional-config-files> \
--wait
```

Where:

`<your-dremio-install-release>` - The name that identifies your Dremio installation. For example, `dremio-1-0`.

`<your-local-path>` - The path to reach your `values-overrides.yaml` configuration file.

(Optional) `version <optional-helm-chart-version>` - The version of Dremio's Helm chart to be used. If not provided, defaults to the latest.

(Optional) `set-file <optional-config-file>` - An optional configuration file for deploying Dremio. For example, an Identity Provider configuration file, which is not defined in the `values-overrides.yaml` and can be provided here through this option.

For OpenShift, the command requires an additional `values` option with the path to the OpenShift-specific `values-openshift-overrides.yaml` configuration file. This additional option must be placed before the `values` option with the `values-overrides.yaml` configuration file, resulting in its substitution first.

Run the following command for OpenShift:

```
helm install <your-dremio-install-release> oci://quay.io/dremio/dremio-helm \
--values <your-local-path1>/values-openshift-overrides.yaml \
--values <your-local-path2>/values-overrides.yaml \
--version <optional-helm-chart-version> \
--set-file <optional-config-files> \
--wait
```

Where:

`<your-dremio-install-release>` - The name that identifies your Dremio installation. For example, `dremio-1-0`.

`<your-local-path1>` - The path to reach your `values-openshift-overrides.yaml` configuration file. Only required for OpenShift.

`<your-local-path2>` - The path to reach your `values-overrides.yaml` configuration file.

(Optional) `version <optional-helm-chart-version>` - The version of Dremio's Helm chart to be used. If not provided, defaults to the latest.

(Optional) `set-file <optional-config-file>` - An optional configuration file for deploying

Dremio. For example, an Identity Provider configuration file, which is not defined in the `values-overrides.yaml` and can be provided here through this option.

Monitor the deployment using the following commands:

Standard Kubernetes

OpenShift

Run the following command for any Kubernetes environment except for OpenShift:

```
kubectl get pods
```

For OpenShift, run the following command:

```
oc get pods
```

When all of the pods are in the `Ready` state, the deployment is complete.

Troubleshooting

If a pod remains in `Pending` state for more than a few minutes, run the following command to view its status to check for issues, such as insufficient resources for scheduling:

```
kubectl describe pods <pod-name>
```

If the events at the bottom of the output mention insufficient CPU or memory, do one of the following:

Adjust the values in the `values-overrides.yaml` configuration file and redeploy.

Add more resources to your Kubernetes cluster.

If a pod returns a failed state (especially `dremio-master-0`, the most important pod), use the following commands to collect the logs:

Standard Kubernetes

OpenShift

Run the following command for any Kubernetes environment except for OpenShift:

```
kubectl logs dremio-master-0
```

For OpenShift, run the following command:

```
oc logs deployment/dremio-master
```

Step 2: Connecting to Dremio

Now that you've installed the Helm chart and deployed Dremio on Kubernetes, the next step is connecting to Dremio, where you have the following options:

Dremio Console

OpenShift Route

BI Tools via ODBC/JDBC

BI Tools via Apache Arrow Flight

To connect to Dremio via [the Dremio console](#), run the following command to use the ``services dremio-client`` in Kubernetes to find the host for the Dremio console:

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
...	AGE
...

If the value in the ``TYPE`` column of the output is ``LoadBalancer``, access the Dremio console through the address in the ``EXTERNAL_IP`` column and port **9047**. For example, in the output below, the value under the ``EXTERNAL-IP`` column is ``8.8.8.8``. Therefore, access the Dremio console through `<http://8.8.8.8:9047>`.

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
dremio-client	LoadBalancer	10.99.227.180	8.8.8.8	31010:32260/TCP,9047:30620/TCP
	2d			

If you want to change the exposed port on the load balancer, change the value of the setting ``coordinator.web.port`` in the file ``values-overrides.yaml``.

If the value in the ``TYPE`` column of the output is ``NodePort``, access the Dremio console through `<http://localhost:30670>`.

To expose Dremio externally using OpenShift Routes, do the following:

```
$ oc expose service dremio-client --port=9047 --name=dremio-ui
```

```
$ oc get route dremio-ui -o jsonpath='{.spec.host}'
```

To connect your BI tools to Dremio via ODBC/JDBC, run the following command to use the ``services dremio-client`` in Kubernetes to find the host for ODBC/JDBC connections by using the following command:

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
...	AGE
...

If the value in the `TYPE` column of the output is `LoadBalancer`, access Dremio using ODBC/JDBC through the address in the `EXTERNAL_IP` column and port **31010**. For example, in the output below, the value under the `EXTERNAL-IP` column is `8.8.8.8`. Therefore, access Dremio using ODBC/JDBC on port 31010 through `<http://8.8.8.8:31010>`.

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
	AGE			
dremio-client	LoadBalancer	10.99.227.180	8.8.8.8	
31010:32260/TCP,9047:30620/TCP 2d				

If you want to change the exposed port on the load balancer, change the value of the setting `coordinator.client.port` in the file `values-overrides.yaml`.

If the value in the `TYPE` column of the output is `NodePort`, access Dremio using ODBC/JDBC through `<http://localhost:32390>`.

To connect your BI tools to Dremio via Apache Arrow Flight, run the following command to use the `services dremio-client` in Kubernetes to find the host for Apache Arrow Flight connections by using the following command:

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
	AGE			
...
	...			

If the value in the `TYPE` column of the output is `LoadBalancer`, access Dremio using Apache Arrow Flight through the address in the `EXTERNAL_IP` column and port **32010**.

For example, in the output below, the value under the `EXTERNAL-IP` column is `8.8.8.8`. Therefore, access Dremio using Apache Arrow Flight through `<http://8.8.8.8:32010>`.

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
	AGE			
dremio-client	LoadBalancer	10.99.227.180	8.8.8.8	
31010:32260/TCP,9047:30620/TCP 2d				

If you want to change the exposed port on the load balancer, change the value of the setting `coordinator.flight.port` in the file `values-overrides.yaml`.

If the value in the `TYPE` column of the output is `NodePort`, access Dremio using Apache Arrow Flight through `<http://localhost:31357>`.

Was this page helpful?

[Previous

Kubernetes Environments](/current/deploy-dremio/kubernetes-environments)[Next

Configuring Your Values](/current/deploy-dremio/configuring-kubernetes/)

Prerequisites

Additional Prerequisites for the Enterprise Edition

Additional Prerequisites for the OpenShift

Step 1: Deploy Dremio

Step 2: Connecting to Dremio

Source:
<https://docs.dremio.com/current/deploy-dremio/configuring-kubernetes/>

Version: current [26.x]

On this page

Configuring Your Values to Deploy Dremio to Kubernetes

Helm is a standard for managing Kubernetes applications, and the Helm chart defines how applications are deployed to Kubernetes. Dremio's Helm chart contains the default deployment configurations, which are specified in the `values.yaml`.

Dremio recommends configuring your deployment values in a separate `.yaml` file since it will allow simpler updates to the latest version of the Helm chart by copying the separate configuration file across Helm chart updates.

FREE TRIAL

If you are using an **Enterprise Edition free trial**, you don't need to do all the configurations described on this page. Instead, follow the configuration steps described in Get Started with the Enterprise Edition Free Trial.

Configure Your Values

To configure your deployment values, do the following:

Get the `values-overrides.yaml` configuration file and save it locally. Click here to download the file.

The `values-overrides.yaml` configuration file

```
# A Dremio License is required
dremio:
  license: "<your-license-key>"
  image:
    repository: quay.io/dremio/dremio-enterprise-jdk21

# Configuration file customization
# The configFiles and configBinaries options provide the ability to override or add
```

```

configuration files
  # included in the Dremio ConfigMap. Both use a map where keys correspond to the
filenames
  # and values are the file contents.

  # configFiles: Use this to provide text-based configuration files that will be mounted
in /opt/dremio/conf/
  # Note: The dremio.conf file is controlled by multiple settings in this values file
and
  # should not be directly overridden here.
  # Example:
#configFiles:
# vault_config.json: |
#   {
#     <your-vault-json-config>
#   }

# configBinaries: Use this to provide binary configuration files (encoded as base64)
# These files will also be mounted in /opt/dremio/conf/
# Example:
#configBinaries:
#  custom-truststore.jks: "base64EncodedBinaryContent"

# dremioConfExtraOptions: Use this to add settings in dremio.conf
# Example:
#dremioConfExtraOptions:
# # Enable SSL for fabric services
# "services.fabric.ssl.enabled": true
# "services.fabric.ssl.auto-certificate.enabled": false

  # Hive 2 and 3 configuration files - can be provided here too. See:
https://docs.dremio.com/current/deploy-dremio/configuring-kubernetes/#hive
#hive2ConfigFiles:
#
#hive3ConfigFiles:
#

# To pull images from Dremio's Quay, you must create an image pull secret. For more
info, see:
#
https://kubernetes.io/docs/concepts/containers/images/#specifying-imagepullsecrets-on-a-pod
# All of the images are pulled using this same secret.
imagePullSecrets:
  - <your-pull-secret-name>

# Dremio Coordinator
coordinator:
  web:
    auth:
      enabled: true
      type: "internal" # Valid types are: internal, ldap, azuread, oauth, oauth+ldap
      # if enabled is true and type ldap, azuread, oauth, or oauth+ldap
      # Uncomment the entry below and provide the JSON configuration inline

```

```

    # OR use --set-file coordinator.web.auth.ssoFile=/path/to/file for the SSO
provider configuration file during Helm install
    # for more information about the file format for your SSO provider
                                                                    # see
https://docs.dremio.com/current/get-started/cluster-deployments/customizing-configuration/dremio-conf/sso-config/
    # ssoFile: |
    # {
    # <your-sso-json-file-content>
    # }
    tls:
      enabled: false
      secret: "<your-tls-secret-name>"
    client:
      tls:
        enabled: false
        secret: "<your-tls-secret-name>"
    flight:
      tls:
        enabled: false
        secret: "<your-tls-secret-name>"
    resources:
      requests:
        cpu: "32"
        memory: "64Gi"
      limits:
        memory: "64Gi"
    volumeSize: 512Gi

# Where Dremio stores metadata, Reflections, uploaded files, and backups. The
distributed store is required for Dremio to be operational.
# For more information, see
https://docs.dremio.com/current/get-started/cluster-deployments/architecture/distributed-storage/
distStorage:
  # The supported distributed storage types are: aws, gcp, or azureStorage. For
S3-compatible storage use aws.
  type: <your-distributed-storage-type> # Add here your distributed storage template
from
http://docs.dremio.com/current/deploy-dremio/configuring-kubernetes/#configuring-the-distributed-storage

# MongoDB is the backing store for the Open Catalog. Backups are enabled by default and
will take place automatically. Dremio will write these backups to your distributed
storage location. Not all authentication types are supported. See our distributed
storage docs link above. Lack of support will be noted where applicable.
mongodb:
  backup:
    enabled: true

# Dremio Catalog
catalog:
  externalAccess:
    enabled: true

```



```

  tls:
    enabled: false
    secret: "<your-catalog-tls-secret-name>"
  # This is where Iceberg tables created in your catalog will reside
  storage:
# The supported catalog storage types are: S3, azure and GCS. For S3-compatible storage
use S3.
    type: <your-catalog-storage-type>
#      Add      here      your      catalog      storage      template      from
https://docs.dremio.com/current/deploy-dremio/configuring-kubernetes/#configuring-storage-for-dremio-catalog

service:
  type: LoadBalancer

```

Edit the `values-overrides.yaml` file to configure your values. See the following sections for details on each configuration option:

License

Pull Secret

Coordinator

Coordinator's Distributed Storage

Open Catalog

Advanced Values Configurations

IMPORTANT

In all code examples, `...` denotes additional values that have been omitted.

Group all values associated with a given parent key in the YAML under a single instance of that parent, for example:

Do

```

dremio:
  key-one: <value-one>
  key-two:
    key-three: <value-two>

```

Do not

```

dremio:
  key-one: <value-one>

dremio:
  key-two:
    key-three: <value-two>

```

Please note the parent relationships at the top of each YAML snippet and subsequent values throughout this section. The hierarchy of keys and indentations in YAML must be respected.

Save the `values-overrides.yaml` file.

Once done with the configuration, deploy Dremio to Kubernetes. See how in [Deploying Dremio to Kubernetes](#).

License

Provide your license key. To obtain a license, see [Licensing](#).

Add this configuration under the parent, as shown in the following example:

Configuration of the license key

```
dremio:
  license: "<your-license-key>"
  ...
```

Pull Secret

Provide the secret used to pull the images from Quay.io as follows:

Log in to [Quay.io](#), select your account name at the top right corner, and select **Account Settings** in the drop-down menu.

Click **Generate Encrypted Password**, type your password, and click **Verify**.

On the next dialog, select **Kubernetes Secret**, and follow steps 1 and 2 to download the secret and run the command to submit the secret to the cluster.

Add the configuration under the parent, as shown in the following example:

Configuration of the secret to pull images from Quay.io

```
imagePullSecrets:
  - <your-quayio-secret-name>
```

Coordinator

Resource Configuration

Configure the volume size, resources limits, and resources requests. To configure these values, see [Recommended Resources Configuration](#).

Add this configuration under the parents, as shown in the following example:

Configuration of the coordinator's resources with example values

```
coordinator:
```

```
resources:
  requests:
    cpu: 15
    memory: 30Gi
  volumeSize: 100Gi
  ...
```

Identity Provider

Optionally, you can configure authentication via an identity provider. Each type of identity provider requires an additional configuration file provided during Dremio's deployment.

Select the authentication `type`, and follow the corresponding link for instructions on how to create the associated configuration file:

`azuread` - See how to [configure Microsoft Entra ID with user and group lookup](#).

`ldap` - See how to [configure Dremio for LDAP](#).

`oauth` - See how to [configure Dremio for OpenID](#).

`oauth+ldap` - See how to [configure Dremio for Hybrid OpenID+LDAP](#).

Add this configuration under the parents, as shown in the following example:

Configuration of the coordinator's identity provider

```
coordinator:
  web:
    auth:
      type: <your-auth-type>
  ...
```

The identity provider configuration file can be embedded in your `values-overrides.yaml`. To do this, use the `ssoFile` option and provide the JSON content constructed per the instructions linked above. Here is an example for Microsoft Entra ID:

Configuration of an embedded identity provider file with an example for Microsoft Entra ID

```
coordinator:
  web:
    auth:
      enabled: true
      type: "azuread"
      ssoFile: |
      {
        "oAuthConfig": {
          "clientId": "<your-client-id>",
          "clientSecret": "<your-secret>",
          "redirectUrl": "<your-redirect-url>",
```

```

    "authorityUrl": "https://login.microsoftonline.com/<your-tenant-id>/v2.0",
    "scope": "openid profile",
    "jwtClaims": {
      "userName": "<your-preferred-username>"
    }
  }
}
...

```

For examples for the other types, see [Identity Providers](#)

This is not the only configuration file that can be embedded inside the `values-overrides.yaml` file. However, these are generally used for advanced configurations. For more information, see [Additional Configuration](#).

Transport Level Security

Optionally enable the desired level of Transport Level Security (TLS) by setting `enabled: true` for client, Arrow Flight, or web TLS. To provide the TLS secret, see [Creating a TLS Secret](#).

Add this configuration under the parent, as shown in the following example:

Configuration of TLS for the coordinator

```

coordinator:
  client:
    tls:
      enabled: false
      secret: <your-tls-secret>
  flight:
    tls:
      enabled: false
      secret: <your-tls-secret>
  web:
    tls:
      enabled: false
      secret: <your-tls-secret>
...

```

note

If Web TLS is enabled, see [Configuring Open Catalog when the Coordinator Web is Using TLS](#).

Coordinator's Distributed Storage

This is where Dremio stores metadata, Reflections, uploaded files, and backups. A distributed store is required for Dremio to be operational. The supported types are Amazon S3 or S3-compatible storage, Azure Storage, and Google Cloud Storage (GCS). For examples of configurations, see [Configuring the Distributed Storage](#).

Add this configuration under the parent, as shown in the following example:

Configuration of the coordinator's distributed storage

```
distStorage:
  type: "<your-dist-store-type>"
  ...
```

Open Catalog

The configuration for the Open Catalog has several options:

Configuring storage for the Open Catalog is mandatory since this is the location where Iceberg tables created in the catalog will be written. For configuring the storage, see [Configuring Storage for the Open Catalog](#).

Add this configuration under the parent, as shown in the following example:

Configuration of the storage for the Open Catalog

```
catalog:
  storage:
    location: <your-object-store-path>
    type: <your-object-store-type>
  ...
```

(Optional) MongoDB is the backing store for Open Catalog. Its backup is enabled by default. This backup is written to distributed storage. Open Catalog backup can be disabled by setting `enabled` to `false`. The configuration shown here performs an automatic Open Catalog backup every day at midnight, and keeps the last three backups.

Enablement of the Open Catalog Backing Store Backup

```
mongodb:
  backup:
    enabled: true
    schedule: "0 0 * * *"
    keep: 3
```

(Optional) Configure external access if you want to connect to the Open Catalog with an engine other than Dremio that supports Iceberg REST. For example, Spark.

Add this configuration under the parent, as shown in the following example:

Configuration of external access for the Open Catalog

```
catalog:
  externalAccess:
    enabled: true
  ...
```

(Optional) Use Transport Level Security (TLS) for external access to require clients connecting to the Open Catalog from outside the namespace to use TLS. To configure it, see [Configuring TLS for Open Catalog External Access](#).

Add this configuration under the parent, as shown in the following example:

Configuration of TLS for external access to the Open Catalog

```
catalog:
  externalAccess:
    enabled: true
    tls:
      enabled: true
      secret: <your-catalog-tls-secret>
  ...
```

(Optional) If Dremio coordinator web access is using TLS, additional configuration is necessary. To configure it, see [Configuring Open Catalog When the Coordinator Web is Using TLS](#).

Add this configuration under the parent, as shown in the following example:

Configuration of the Open Catalog when the coordinator web access is using TLS

```
catalog:
  externalAccess:
    enabled: true
    authentication:
      authServerHostname: <your-auth-server-host>
  ...
```

Save the `values-overrides.yaml` file.

Once done with the configuration, deploy Dremio to Kubernetes. See how in the topic [Deploying Dremio to Kubernetes](#).

Configuring Your Values - Advanced

OpenShift

warning

OpenShift has additional prerequisites that must be applied before installing Dremio. For more information, see [Deploy on Kubernetes - Prerequisites](#).

To deploy successfully on OpenShift, you must deploy with two override files. The YAML file you've been using to this point (`values-overrides.yaml`), and an additional YAML file mentioned below (`openshift-overrides.yaml`) with security settings required by OpenShift per its default configuration. Both can be provided in a single Helm install command.

Get the `openshift-overrides.yaml` configuration file and save it locally.

[Click here](#) to download the file.

Dremio Platform Images

The Dremio platform requires 18 images when running fully featured. All images are published by Dremio to our Quay and are listed below. If you want to use a private mirror of our repository, add the snippets below to `values-overrides.yaml` to repoint to your own.

Dremio Platform Images

note

If creating a private mirror, use the same repository names and tags from [Dremio's Quay.io](#).

This is important for supportability.

```
dremio:
  image:
    repository: quay.io/dremio/dremio-enterprise-jdk21
    tag: <the-image-tag-from-quayio>
```

```
busyBox:
  image:
    repository: quay.io/dremio/busybox
    tag: <the-image-tag-from-quayio>
```

```
k8s:
  image:
    repository: quay.io/dremio/alpine/k8s
    tag: <the-image-tag-from-quay-io>
```

```
engine:
  operator:
    image:
      repository: quay.io/dremio/dremio-engine-operator
      tag: <the-image-tag-from-quay-io>
```

```
zookeeper:
  image:
    repository: quay.io/dremio/zookeeper
    tag: <the-image-tag-from-quay-io>
```

```
opensearch:
  image:
    repository: quay.io/dremio/dremio-search-opensearch
    tag: <the-image-tag-from-quay-io> # The tag version must be a valid OpenSearch
version as listed here https://opensearch.org/docs/latest/version-history/
preInstallJob:
```

```
image:
  repository: quay.io/dremio/dremio-search-init
  tag: <the-image-tag-from-quay-io>
```

```
opensearchOperator:
  manager:
    image:
      repository: quay.io/dremio/dremio-opensearch-operator
      tag: <the-image-tag-from-quay-io>
  kubeRbacProxy:
    image:
      repository: quay.io/dremio/kubebuilder/kube-rbac-proxy
      tag: <the-image-tag-from-quay-io>
```

```
mongodbOperator:
  image:
    repository: quay.io/dremio/dremio-mongodb-operator
    tag: <the-image-tag-from-quay-io>
```

```
mongodb:
  image:
    repository: quay.io/dremio/percona/percona-server-mongodb
    tag: <the-image-tag-from-quay-io>
```

```
catalogservices:
  image:
    repository: quay.io/dremio/dremio-catalog-services-server
    tag: <the-image-tag-from-quay-io>
```

```
catalog:
  image:
    repository: quay.io/dremio/dremio-catalog-server
    tag: <the-image-tag-from-quay-io>
  externaAccess:
    image:
      repository: quay.io/dremio/dremio-catalog-server-external
      tag: <the-image-tag-from-quay-io>
```

```
nats:
  container:
    image:
      repository: quay.io/dremio/nats
      tag: <the-image-tag-from-quay-io>
  reloader:
    image:
      repository: quay.io/dremio/natsio/nats-server-config-reloader
      tag: <the-image-tag-from-quay-io>
  natsBox:
```



```
container:
  image:
    repository: quay.io/dremio/natsio/nats-box
    tag: <the-image-tag-from-quay-io>
```

```
telemetry:
  image:
    repository: quay.io/dremio/otel/opentelemetry-collector-contrib
    tag: <the-image-tag-from-quay-io>
```

Scale-out Coordinators

Dremio can scale to support high-concurrency use cases through scaling coordinators. Multiple stateless coordinators rely on the primary coordinator to manage Dremio's state, enabling Dremio to support many more concurrent users. These scale-out coordinators are intended for high query throughput and are not applicable for standby or disaster recovery. While scale-out coordinators generally reduce the load on the primary coordinator, the primary coordinator's vCPU request should be increased for every two scale-outs added to avoid negatively impacting performance.

Perform this configuration in this section of the file, where count refers to the number of scale-outs. A count of 0 will provision only the primary coordinator:

Configuration of scale-out coordinators with an example value

```
coordinator:
  count: 1
  ...
```

note

When using scale-out coordinators, the load balancer session affinity should be enhanced. See: [Advanced Load Balancer Configuration](#).

Configuring Kubernetes Pod Metadata (including Node Selector)

It's possible to add metadata both globally and to each of the StatefulSets (coordinators, classic engines, ZooKeeper, etc.), including configuring a node selector for pods to use specific node pools.

warning

Define these values with caution and foreknowledge of expected entries because any misconfiguration may result in Kubernetes being unable to schedule your pods.

Use the following options to add metadata:

`~labels:` - Configured using key-value pairs as shown in the following examples:

Configuration of a global label with a key-value example

```
labels:
  foo: bar
```

Configuration of a StatefulSet label for the Open Catalog with a key-value example

```
catalog:
  labels:
    foo: bar
  ...
```

For more information on labels, see the Kubernetes documentation on [Labels and Selectors](#).

`annotations:` - Configured using key-value pairs as shown in the following examples.

Configuration of a global annotation with a key-value example

```
annotations:
  foo: bar
```

Configuration of a StatefulSet annotation for MongoDB with a key-value example

```
mongodb:
  annotations:
    foo: bar
  ...
```

For more information on annotations, see the Kubernetes documentation on [Annotations](#).

`tolerations:` - Configured using a specific structure as shown in the following examples:

Configuration of a global toleration with example values

```
tolerations:
- key: "key1"
  operator: "Equal"
  value: "value1"
  effect: "NoSchedule"
```

Configuration of a StatefulSet toleration for the Open Catalog with example values

```
catalog:
  tolerations:
  - key: "key1"
    operator: "Equal"
    value: "value1"
    effect: "NoSchedule"
  ...
```

For more information on tolerations, see the Kubernetes documentation on [Taints and Tolerations](#).

`nodeSelector:` - Configured using a specific structure as shown in the following examples.

Configuration of a global node selector with an example value

```
nodeSelector:
  nodetype: coordinator
```

Configuration of a StatefulSet node selector for the coordinator with an example value

```
coordinator:
  nodeSelector:
    nodetype: coordinator
  ...
```

To understand the structure and values to use in the configurations, expand "Metadata Structure and Values" below:

Metadata Structure and Values

For global metadata:

Global metadata structure

```
annotations: {}
labels: {}
tolerations: []
nodeSelector: {}
```

For StatefulSet metadata:

StatefulSet metadata structure for the coordinator

```
coordinator:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: coordinator
```

StatefulSet metadata structure for the executors

```
executor:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: coordinator
```

StatefulSet metadata structure for the Open Catalog

```
catalog:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: catalog
```

StatefulSet metadata structure for the Open Catalog services

```
catalogservices:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: catalogservices
```

StatefulSet metadata structure for MongoDB

```
mongodb:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: mongo
```

StatefulSet metadata structure for OpenSearch

```
opensearch:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: operators
oidcProxy:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodeType: utils
preInstallJob:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodeType: jobs
```

StatefulSet metadata structure for NATS

```
nats:
```

```
podTemplate:
  merge:
    spec:
      annotations: {}
      labels: {}
      tolerations: []
      nodeSelector:
        nodetype: nats
```

StatefulSet metadata structure for the MongoDB operator

```
mongodb0operator:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: operators
```

StatefulSet metadata structure for the OpenSearch operator

```
opensearch0operator:
  annotations: {}
  labels: {}
  tolerations: []
  nodeSelector:
    nodetype: operators
```

Configuring Pods Priority

You can configure the priority of Dremio pods through priority classes. First, define the priority class, as shown in the following example:

Definition of a `high-priority` priority class

```
apiVersion: scheduling.k8s.io/v1
kind: PriorityClass
metadata:
  name: high-priority
value: 1000000
globalDefault: false
description: "This priority class should be used for coordinator pods only."
```

Then, apply the priority class under the parents, as shown in the following example:

Configuration of the `high-priority` priority class for the coordinator

```
coordinator:
  priorityClassName: high-priority
```

To understand the structure and values to use in the configurations, expand "Priority Class Configuration Structure and Values" below:

Priority Class Configuration Structure and Values

Priority class configuration for the coordinator

```
coordinator:  
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for the Open Catalog

```
catalog:  
  priorityClassName: <your-priority-class-name>  
  externalAccess:  
    priorityClassName: <your-priority-class-name>
```

Priority class configuration for the Open Catalog services

```
catalogservices:  
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for the engine

```
engine:  
  executor:  
    priorityClassName: <your-priority-class-name>  
  operator:  
    priorityClassName: <your-priority-class-name>
```

Priority class configuration for OpenSearch

```
opensearch:  
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for the OpenSearch operator

```
opensearchOperator:  
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for MongoDB

```
mongodb:  
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for the MongoDB hooks

```
mongodbHooks:  
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for NATS

```
nats:
  podTemplate:
    merge:
      spec:
        priorityClassName: <your-priority-class-name>
  natsBox:
    podTemplate:
      merge:
        spec:
          priorityClassName: <your-priority-class-name>
```

Priority class configuration for ZooKeeper

```
zookeeper:
  priorityClassName: <your-priority-class-name>
```

Priority class configuration for telemetry

```
telemetry:
  priorityClassName: <your-priority-class-name>
```

note

To verify which priority class is applied to each pod, run the command below, and check the `PRIORITY_CLASS` column:

Run kubectl to list the pods and their priority class

```
kubectl get pods -o
custom-columns="NAME:.metadata.name,PRIORITY_CLASS:.spec.priorityClassName" -n dremio
```

Configuring Extra Environment Variables

Optionally, you can define extra environment variables to be passed to either coordinators or executors. This can be done by adding the configuration under the parents as shown in the following examples:

Configuration of extra environment variables for the coordinator

```
coordinator:
  extraEnvs:
    - name: <your-variable-name>
      value: "<your-variable-value>"
  ...
```

Configuration of extra environment variables for the executors

```
executor:
```

```
extraEnvs:
  - name: <your-variable-name>
    value: "<your-variable-value>"
  ...
```

Environment variables defined as shown will be applied to Executors of both [Classic Engines](#) and [New Engines](#).

Advanced Load Balancer Configuration

Dremio will create a public load balancer by default, and the Dremio Client service will provide an external IP to connect to Dremio. For more information, see [Connecting to the Dremio Console](#).

Private Cluster - For private Kubernetes clusters (no public endpoint), set ``internalLoadBalancer: true``. Add this configuration under the parent as shown in the following example:

Configuration of an internal load balancer

```
service:
  type: LoadBalancer
  internalLoadBalancer: true
  ...
```

Static IP - To define a static IP for your load balancer, set ``loadBalancerIP: <your-static-IP>``. If unset, an available IP will be assigned upon creation of the load balancer. Add this configuration under the parent as shown in the following example:

Configuration of a static IP for the load balancer

```
service:
  type: LoadBalancer
  loadBalancerIP: <your-desired-ip>
  ...
```

tip

This can be helpful if DNS is configured to expect Dremio to have a specific IP.

Session Affinity - If leveraging scale-out coordinators, set this to ``ClientIP``, otherwise leave unset. Add this configuration under the parent as shown in the following example:

Configuration of session affinity for scale-out coordinators

```
service:
  type: LoadBalancer
  sessionAffinity: ClientIP
  ...
```


Additional Load Balancer Configuration for Amazon EKS in Auto Mode

If deploying Dremio to Amazon EKS (Elastic Kubernetes Service) in Auto Mode, you need to add service annotations for the load balancer to start (for more information, see [Use Service Annotations to configure Network Load Balancers](#)). Add this configuration under the parent as shown in the following example:

Configuration of service annotations for Amazon EKS in Auto Mode

```
service:
  type: LoadBalancer
  annotations:
    service.beta.kubernetes.io/aws-load-balancer-scheme: internet-facing
  ...
```

Advanced TLS Configuration for OpenSearch

Dremio generates Transport Level Security (TLS) certificates by default for OpenSearch, and they are rotated monthly. However, if you want to have your own, you need to create two secrets containing the relevant certificates. The format of the secrets is different from the other TLS secrets shown on this page, and the `tls.crt`, `tls.key`, and `ca.crt` files must be in PEM format. Use the example below as a reference to create your secrets:

Run `kubectcl` to create two secrets for your own TLS certificates for OpenSearch

```
kubectcl create secret generic opensearch-tls-certs \
  --from-file=tls.crt --from-file=tls.key --from-file=ca.crt

kubectcl create secret generic opensearch-tls-certs-admin \
  --from-file=tls.crt --from-file=tls.key --from-file=ca.crt
```

Add the snippet below to the `values-overrides.yaml` file before deploying Dremio. Because OpenSearch requires TLS, if certificate generation is disabled, you must provide a certificate.

Configuration of TLS certificates for OpenSearch

```
opensearch:
  tlsCertsSecretName: <opensearch-tls-certs>
  disableTlsCertGeneration: true
  ...
```

Advanced Configuration of Engines

Dremio's default resource offset is `reserve-2-8`, where the first value represents 2 vCPUs and the second represents 8 GB of RAM. If you need to change this default for your created engines, add the following snippet to `values-overrides.yaml` and set the `defaultOffset` to one of the configurable offsets listed below, which are available out

of the box:

`reserve-0-0`

`reserve-2-4`

`reserve-2-8`

`reserve-2-16`

The listed values are keys and thus must be provided in this exact format in the snippet below.

Configuration of the default resource offset for engines with an example value

```
engine:
  options:
    resourceAllocationOffsets:
      defaultOffset: reserve-2-8
  ...
```

Configuration of Classic Engines

note

You should only use classic engines if the new ones introduced in Dremio 26.0 are not appropriate for your use case. Classic and new engines are not intended to be used side by side.

Classic engines will not auto-start/auto-stop, which is only possible with the new engines.

The classic way of configuring engines is still supported, and you can add this snippet to `values-overrides.yaml` as part of the deployment. Note that this snippet is a configuration example, and you should adjust the values to your own case.

Configuration of classic engines with example values

```
executor:
  resources:
    requests:
      cpu: "16"
      memory: "120Gi"
    limits:
      memory: "120Gi"
  engines: ["default"]
  count: 3
  volumeSize: 128Gi
  cloudCache:
    enabled: true
  volumes:
    - size: 128Gi
  ...
```

Engine Overrides

Engine overrides are primarily used in conjunction with classic engines to modify the configuration of one or more named engines. By default, every engine inside the ``engines`` list under ``executor`` will be the same. The values set under ``executor`` act as the default for all engines. Thus, the engine overrides do not need to be exhaustive.

Configuration of overrides for an engine named 'small'

```
engineOverride:
  small:
    cpu: "8"
    memory: "60Gi"
    count: 2
    cloudCache:
      enabled: false
```

Engine overrides can also be used with the new engines, but only to disable the Cloud Columnar Cache (C3) option. C3 is enabled by default on all new engines, but you can choose to disable it if needed.

Telemetry

Telemetry egress is enabled by default. These metrics provide visibility into various components and services, ensuring optimal performance and reliability. To disable egress, add the following to your ``values-override.yaml``:

Configuration to disable telemetry

```
telemetry:
  enabled: false
  ...
```

Logging

By default, Dremio enables logging with a pre-defined volume size, which you can check in the ``values.yaml`` file by downloading Dremio's Helm chart. To override the default configuration, add the following to your ``values-overrides.yaml``:

Configuration of logging

```
dremio:
  log:
    enabled: true
    volume:
      size: 10Gi
      storageClass: ""
  ...
```

Disabling Parts of the Deployment

You can disable some components of the Dremio platform if their functionality does not pertain to your use case. Dremio's functionality will continue to work if any of these components described in this section are disabled.

Semantic Search

To disable Semantic Search, add this configuration under the parent as shown in the following example:

Configuration to disable Semantic Search

```
opensearch:
  enabled: false
  replicas: 0
```

Additional Configuration

Dremio has several configuration and binary files to define the behavior for enabling authentication via an identity provider, logging, connecting to Hive, etc. During the deployment, these files are combined and used to create a [Kubernetes ConfigMap](#). This ConfigMap is, in turn, used by the Dremio deployment as the source of truth for various settings. Options can be used to embed these in the `values-override.yaml` configuration file.

To inspect Dremio's configuration files or perform a more complex operation not shown here, see [Downloading Dremio's Helm Charts](#).

Additional Config Files

Use the `configFiles` option to add configuration files to your Dremio deployment. You can add multiple files, each of which is a key-value pair. The key is the file name, and the value is the file content. These can be TXT, XML, or JSON files. For example, here is how to embed the configuration for Hashicorp Vault, followed by a separate example file:

Configuration of additional configuration files with example JSONs

```
dremio:
  configFiles:
    vault_config.json: |
      {
        "vaultUrl": "https://your-vault.com",
        "namespace": "optional/dremio/global/vault/namespace",
        "auth": {
          "kubernetes": {
            "vaultRole": "dremio-vault-role",
            "serviceAccountJwt": "file:///optional/custom/path/to/serviceAccount/jwt",
            "loginMountPath": "optional/custom/kubernetes/login/path"
          }
        }
      }
```

```

    }
  }
  another_config.json: |
  {
    "key-in-this-file": "content-of-this-key"
  }
  ...

```

Additional Config Variables

Use the ``dremioConfExtraOptions`` option to add new variables to your Dremio deployment. For example, here is how to enable Transport Layer Security (TLS) between executors and coordinators, leveraging auto-generated self-signed certificates.

Configuration of additional configuration variables with an example to enable TLS

```

dremio:
  dremioConfExtraOptions:
    "services.fabric.ssl.enabled": true
    "services.fabric.ssl.auto-certificate.enabled": true
  ...

```

Additional Java Truststore

Use the ``trustStore`` option under ``advancedConfigs`` to provide the password and content of a Java truststore file. The content must be base64-encoded. To extract the encoded content, you can use ``cat truststore.jks | base64``. Add this configuration under the parents as shown in the following example:

Configuration of an additional Java truststore with a truststore password

```

dremio:
  advancedConfigs:
    trustStore:
      enabled: true
      password: "<your-truststore-password>"
      binaryData: "base64EncodedBinaryContent"

```

Additional Config Binary Files

Use the ``configBinaries`` option to provide binary configuration files. Provided content must be base64-encoded. Add this configuration under the parents as shown in the following example:

Configuration of additional binary configuration files

```

dremio:
  configBinaries:
    custom-binary.conf: "base64EncodedBinaryContent"

```

...

Hive

Use the `hive2ConfigFiles` option to configure Hive 2. Add this configuration under the parents as shown in the following example:

Configuration of Hive 2 with an example for the `hive-site.xml` file

```
dremio:
  hive2ConfigFiles:
    hive-site.xml: |
      <?xml version="1.0" encoding="UTF-8"?>
      <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
      <configuration>
        <property>
          <n>hive.metastore.uris</n>
          <value>thrift://hive-metastore:9083</value>
        </property>
      </configuration>
    ...
```

Use the `hive3ConfigFiles` option to configure Hive 3. Add this configuration under the parents as shown in the following example:

Configuration of Hive 3 with an example for the `hive-site.xml` file

```
dremio:
  hive3ConfigFiles:
    hive-site.xml: |
      <?xml version="1.0" encoding="UTF-8"?>
      <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
      <configuration>
        <property>
          <n>hive.metastore.uris</n>
          <value>thrift://hive3-metastore:9083</value>
        </property>
      </configuration>
    ...
```

References

Recommended Resources Configuration

The table in this section contains the recommended values for resources requests and volume size to configure Dremio components. In the `values-overrides.yaml` file, set the following values:

Configuration of resources in Dremio components

```
resources:
  requests:
    memory: # Refer to the Memory column in the tables below for recommended values
    cpu: # Refer to the CPU column in the tables below for recommended values
    volumeSize: # Refer to the Volume Size column in the tables below for recommended values
```

Dremio recommends the following configuration values:

Production Configuration

Minimal Configuration

Dremio recommends the following configuration values to operate in a production environment:

Dremio Component	Memory	CPU	Volume Size	Pod Count
Coordinator	64Gi	32	512Gi	1
Catalog Server	8Gi	4	-	1
Catalog Server (External)	8Gi	4	-	1
Catalog Service Server	8Gi	4	-	1
Engine Operator	1Gi	1	-	1
OpenSearch	16Gi	2	100Gi	3
MongoDB	4Gi	8	512Gi ¹	3
NATS	1Gi	700m	-	3
ZooKeeper	1Gi	500m	-	3
Open Telemetry	1Gi	1	-	1
M Engine	120Gi	16	521Gi	4

¹ You can use a smaller volume size if you do not heavily use Iceberg.

The following configuration will deploy a functional Dremio Platform, sized to fit onto a more modest cluster. It is appropriate for a single user to check out Dremio's various features, leveraging our sample data set. For any multi-user and performance-oriented evaluation, the Production Configuration should be used.

Dremio Component	Memory	CPU	Volume Size	Pod Count
Coordinator	8Gi	2	20Gi	1
Catalog Server	1Gi	1	-	1
Catalog Server (External)	1Gi	1	-	1
Catalog Service Server	1Gi	1	-	1
Engine Operator	1Gi	1	-	1
OpenSearch	3Gi	1500m	10Gi	3
MongoDB	1Gi	1	10Gi	3
NATS	1Gi	700m	-	3
ZooKeeper	1Gi	500m	-	1
Open Telemetry	1Gi	1	-	1
XS Engine	8Gi	2	20Gi	1

Expand the widget below for Dremio platform components resource YAML snippets:

Dremio Platform Resource Configuration YAML

Coordinator

```
coordinator:
  resources:
    requests:
      cpu: "32"
      memory: "64Gi"
    limits:
      memory: "64Gi"
  volumeSize: "512Gi"
```

Catalog Server

```
catalog:
  requests:
    cpu: "4"
    memory: "8Gi"
  limits:
    cpu: "4"
    memory: "8Gi"
```

Catalog Service Server

```
catalogservices:
  resources:
    requests:
      cpu: "4"
      memory: "8Gi"
    limits:
      cpu: "4"
      memory: "8Gi"
```

OpenSearch

```
opensearch:
  resources:
    requests:
      memory: "16Gi"
      cpu: "2"
    limits:
      memory: "16Gi"
      cpu: "2"
```

MongoDB

```
mongodb:
  resources:
    requests:
      cpu: "2"
      memory: "2Gi"
    limits:
```



```
    cpu: "4"
    memory: "2Gi"
  storage:
    resources:
      requests:
        storage: "512Gi"
```

NATS

```
nats:
  resources:
    requests:
      cpu: "500m"
      memory: "1024Mi"
    limits:
      cpu: "750m"
      memory: "1536Mi"
```

ZooKeeper

```
zookeeper:
  resources:
    requests:
      cpu: "500m"
      memory: "1Gi"
    limits:
      memory: "1Gi"
  volumeSize: "10Gi"
```

Open Telemetry

```
telemetry:
  resources:
    requests:
      cpu: "1"
      memory: "1Gi"
    limits:
      cpu: "2"
      memory: "2Gi"
```

Creating a TLS Secret

If you have enabled Transport Layer Security (TLS) in your `values-overrides.yaml`, the corresponding secrets must be created before deploying Dremio. To create a secret, run the following command:

Run `kubectl` to create a TLS secret

```
kubectl create secret tls <your-tls-secret-name> --key privkey.pem --cert cert.pem
```

For more information, see [kubectl create secret tls](#) in the Kubernetes documentation.

caution

TLS for OpenSearch requires a secret of a different makeup. See [Advanced TLS Configuration for OpenSearch](#).

Configuring the Distributed Storage

Dremio's distributed store uses scalable and fault-tolerant storage, and it is configured as follows:

In the `values-overrides.yaml` file, find the section with `distStorage:` and `type:`

Configuration of the distributed storage

```
distStorage:
  type: "<your-dist-store-type>"
  ...
```

In `type:`, configure your storage provider with one of the following values:

`"aws"` - For Amazon S3 or S3-compatible storage.

`"azureStorage"` - For Azure Storage.

`"gcp"` - For Google Cloud Storage (GCS) in Google Cloud Platform (GCP).

Select the tab below for the storage provider you chose in step 2, and follow the example to configure your distributed storage:

note

Distributed storage is also used to store Open Catalog backups. You may be required to provide two authentication methods to enable storage of these backups.

Amazon S3 and S3-Compatible

Azure Storage

Google Cloud Storage

For Amazon S3 and S3-Compatible, select the tab below for your type of authentication:

Metadata

Access Key

AWS Profile

EKS Pod Identity

Dremio uses the Identity and Access Management (IAM) role to retrieve the credentials to authenticate. Metadata is only supported in Amazon Web Services Elastic Kubernetes Service (AWS EKS) and requires that the EKS worker node IAM role is configured with sufficient access rights.

Add the configuration under the parent as shown in the following example:

Metadata authentication for the distributed storage

```
distStorage:
  type: "aws"
  aws:
    bucketName: "<your-bucket-name>"
    path: "/"
    authentication: "metadata"
    region: "<your-bucket-region>"
    #
    # Extra Properties
    # Use the extra properties block to provide additional parameters
    # to configure the distributed storage in the generated core-site.xml file.
    #
    #extraProperties: |
    # <property>
    #   <name>the-property-name</name>
    #   <value>the-property-value</value>
    # </property>
```

Where:

`bucketName` - The name of your S3 bucket for distributed storage.

`path` - The path relative to your bucket to create Dremio's directories.

`authentication` - Set as `"metadata"`.

`region` - The AWS region in which your bucket resides. Required even if using S3-compatible.

`extraProperties` - Additional parameters to configure the distributed storage in the generated `core-site.xml` file. Important for S3-compatible and customer-managed KMS encryption.

Dremio uses a configured Amazon Web Services (AWS) Access Key and Secret to authenticate.

Add the configuration under the parent as shown in the following example:

Access Key authentication for the distributed storage

```
distStorage:
  type: "aws"
  aws:
    bucketName: "<your-bucket-name>"
    path: "/"
    authentication: "accessKeySecret"
    region: "<your-bucket-region>"
    credentials:
      accessKey: "<your-access-key>"
      secret: "<your-access-key-secret>"
    #
    # Extra Properties
```

```
# Use the extra properties block to provide additional parameters
# to configure the distributed storage in the generated core-site.xml file.
#
#extraProperties: |
# <property>
#   <name>the-property-name</name>
#   <value>the-property-value</value>
# </property>
```

Where:

`bucketName` - The name of your S3 bucket for distributed storage.

`path` - The path relative to your bucket to create Dremio's directories.

`authentication` - Set as `"accessKeySecret"`.

`region` - The AWS region in which your bucket resides. Required even if using S3-compatible.

`credentials` - The credentials configuration:

`accessKey` - Your AWS access key ID.

`secret` - Your AWS access key secret.

`extraProperties` - Additional parameters to configure the distributed storage in the generated `core-site.xml` file. Important for S3-compatible and customer-managed KMS encryption.

Dremio uses the default Amazon Web Services (AWS) profile to retrieve the credentials to authenticate.

note

You need to add an AWS Access Key to store Open Catalog backups.

Add the configuration under the parent as shown in the following example:

AWS profile authentication for the distributed storage

```
distStorage:
  type: "aws"
  aws:
    bucketName: "<your-bucket-name>"
    path: "/"
    authentication: "awsProfile"
    region: "<your-bucket-region>"
    credentials:
      awsProfileName: "default"
      #accessKey: "<your-access-key>" for Open Catalog Backup
      #secret: "<your-access-key-secret>" for Open Catalog Backup
    #
    # Extra Properties
    # Use the extra properties block to provide additional parameters to configure the
distributed
```

```
# storage in the generated core-site.xml file.
#
#extraProperties: |
# <property>
#   <name>the-property-name</name>
#   <value>the-property-value</value>
# </property>
```

Where:

`bucketName` - The name of your S3 bucket for distributed storage.

`path` - The path relative to your bucket to create Dremio's directories.

`authentication` - Set as `"awsProfile"`.

`region` - The AWS region your bucket resides in. Required even if using S3-compatible.

`credentials` - The credentials configuration:

`awsProfileName` - Set as `"default"`.

`accessKey` - AWS access key ID for Open Catalog backup storage.

`secret` - AWS access key secret for Open Catalog backup storage.

`extraProperties` - Additional parameters to configure the distributed storage in the generated `core-site.xml` file. Important for S3-compatible and customer-managed KMS encryption.

EKS Pod Identities allow for Kubernetes service accounts to be associated with an IAM role. Dremio, in turn, can use this IAM role to retrieve the credentials to authenticate. As both the coordinators and engines require access to distributed storage, both of their `ServiceAccounts` must be associated with an IAM role with sufficient access rights. By default, their `ServiceAccounts` are `dremio-coordinator`, `dremio-engine-executor` for New Engines, and (optional) `dremio-executor` for Classic Engines.

Add the configuration under the parent as shown in the following example:

AWS profile authentication for the distributed storage

```
distStorage:
  type: "aws"
  aws:
    bucketName: "<your-bucket-name>"
    path: "/"
    authentication: "podIdentity"
    region: "<your-bucket-region>"
  #
  # Extra Properties
  # Use the extra properties block to provide additional parameters to configure the
distributed
  # storage in the generated core-site.xml file.
  #
  #extraProperties: |
```

```
# <property>
#   <name>the-property-name</name>
#   <value>the-property-value</value>
# </property>
```

Where:

`bucketName` - The name of your S3 bucket for distributed storage.

`path` - The path relative to your bucket to create Dremio's directories.

`authentication` - Set as `"podIdentity"`.

`region` - The AWS region your bucket resides. Required even if using S3-Compatible.

`extraProperties` - Additional parameters to configure the distributed storage in the generated `core-site.xml` file. Important for S3-compatible and customer-managed KMS encryption.

Extra Properties

Example extra properties for S3-compatible storage and for providing a customer-managed KMS key for an encrypted bucket.

S3-Compatible extra properties

```
extraProperties: |
  <property>
    <name>fs.s3a.endpoint</name>
    <value>0.0.0.0</value>
  </property>
  <property>
    <name>fs.s3a.path.style.access</name>
    <value>true</value>
  </property>
  <property>
    <name>dremio.s3.compat</name>
    <value>true</value>
  </property>
  <property>
    <name>fs.s3a.connection.ssl.enabled</name>
    <value>false</value>
  </property>
```

Customer-managed KMS extra properties

```
extraProperties: |
  <property>
    <name>fs.s3a.connection.ssl.enabled</name>
    <value>true</value>
  </property>
  <property>
    <name>fs.s3a.server-side-encryption-algorithm</name>
    <value>SSE-KMS</value>
```

```

</property>
<property>
  <name>fs.s3a.server-side-encryption.key</name>
  <value>KEY_ARN</value>
</property>

```

For Azure Storage, select the tab below for your type of authentication:

Access Key

Entra ID

Dremio uses the configured Azure Storage account access key to authenticate.

Add the configuration under the parent as shown in the following example:

Access Key authentication for the distributed storage

```

distStorage:
  type: "azureStorage"
  azureStorage:
    accountName: "<your-account-name>"
    authentication: "accessKey"
    filesystem: "<your-blob-container>"
    path: "/"
    credentials:
      accessKey: "<your-access-key>"
  #
  # Extra Properties
  # Use the extra properties block to provide additional parameters to configure the
distributed
  # storage in the generated core-site.xml file.
  #
  #extraProperties: |
  # <property>
  #   <name>the-property-name</name>
  #   <value>the-property-value</value>
  # </property>

```

Where:

`accountName` - The name of your storage account.

`authentication` - Set as `"accessKey"`.

`filesystem` - The name of your blob container to use within the storage account.

`path` - The path relative to the filesystem to create Dremio's directories.

`credentials` - The credentials configuration:

`accessKey` - Your Azure Storage account access key.

`extraProperties` - Additional parameters to configure the distributed storage in the

generated `core-site.xml` file.

Dremio uses the configured Azure client ID (application ID), Microsoft Entra ID token endpoint, and Azure client secret (application password) to authenticate.

note

You need to add an Azure Access Key to store Dremio Catalog backups.

Add the configuration under the parent as shown in the following example:

Entra ID authentication for the distributed storage

```
distStorage:
  type: "azureStorage"
  azureStorage:
    accountName: "<your-account-name>"
    authentication: "entraID"
    filesystem: "<your-blob-container>"
    path: "/"
    credentials:
      clientId: "<your-application-client-id>"
      tokenEndpoint: "<your-token-endpoint>"
      clientSecret: "<your-client-secret>"
      #accessKey: "<your-access-key>" for Open Catalog Backup.
  #
  # Extra Properties
  # Use the extra properties block to provide additional parameters to configure the
distributed
  # storage in the generated core-site.xml file.
  #
  #extraProperties: |
  # <property>
  #   <name>the-property-name</name>
  #   <value>the-property-value</value>
  # </property>
```

Where:

`accountName` - The name of your storage account.

`authentication` - Set as `"entraID"`.

`filesystem` - The name of your blob container to use within the storage account.

`path` - The path relative to the filesystem to create Dremio's directories.

`credentials` - The credentials configuration:

`clientId` - Your Azure client ID (application ID).

`tokenEndpoint` - Your Microsoft Entra ID token endpoint.

`clientSecret` - Your Azure client secret (application password).

`accessKey` - Your access key for Open Catalog Backup.

`extraProperties` - Additional parameters to configure the distributed storage in the generated `core-site.xml` file.

Extra Properties

Example extra properties to configure the Azure Storage data source to access data on the Azure Government Cloud platform.

Azure Government Cloud endpoint extra properties

```
extraProperties: |
  <property>
    <name>fs.azure.endpoint</name>
    <description>The azure storage endpoint to use.</description>
    <value>dfs.core.usgovcloudapi.net</value>
  </property>
```

For Google Cloud Storage (GCS) in Google Cloud Platform (GCP), select the tab below for your type of authentication:

Automatic

Service Account

Dremio uses Google Application Default Credentials to authenticate. This is platform-dependent and may not be available in all Kubernetes clusters.

note

You need to add a service account key to store Open Catalog backups.

Add the configuration under the parent as shown in the following example:

```
distStorage:
  type: "gcp"
  gcp:
    bucketName: "<your-bucket-name>"
    path: "/"
    authentication: "auto"
    #credentials: for Open Catalog backup.
    # clientEmail: "<your-email-for-the-service-account>"
    # privateKey: |-
    #     -----BEGIN PRIVATE KEY-----\n <your-full-private-key-value> \n-----END PRIVATE
KEY-----\n
```

Where:

`bucketName` - The name of your GCS bucket for distributed storage.

`path` - The path relative to the bucket to create Dremio's directories.

`authentication` - Set as `"auto"`.

`credentials` - The credentials configuration, for Open Catalog backup:

`clientEmail` - Your email for the service account that has access to the GCS bucket, for Open Catalog backup.

`privateKey` - Your full private key value, for Open Catalog backup.

Dremio uses a JSON key file generated from the GCP console to authenticate.

Add the configuration under the parent as shown in the following example:

```
distStorage:
  type: "gcp"
  gcp:
    bucketName: "<your-bucket-name>"
    path: "/"
    authentication: "serviceAccountKeys"
    credentials:
      projectId: "<your-project-id>"
      clientId: "<your-client-id>"
      clientEmail: "<your-email-for-the-service-account>"
      privateKeyId: "<your-private-key-id>"
      privateKey: |-
        -----BEGIN PRIVATE KEY-----\n <your-full-private-key-value> \n-----END PRIVATE
KEY-----\n
```

Where:

`bucketName` - The name of your GCS bucket for distributed storage.

`path` - The path relative to your bucket to create Dremio's directories.

`authentication` - Set as `"serviceAccountKeys"`.

`credentials` - The credentials configuration:

`projectId` - Your GCP Project ID that the GCS bucket belongs to.

`clientId` - Your Client ID for the service account that has access to the GCS bucket.

`clientEmail` - Your email for the service account that has access to the GCS bucket.

`privateKeyId` - Your private key ID for the service account that has access to GCS bucket.

`privateKey` - Your full private key value.

note

When using a GCS bucket on Google Kubernetes Engine (GKE), we recommend enabling **Workload Identity** and configuring a Kubernetes service account for Dremio with an associated workload identity that has access to the GCS bucket.

Configuring Storage for the Open Catalog

To use the Open Catalog, configure the storage settings based on your storage provider (for example, Amazon S3, Azure Storage, or Google Cloud Storage). This configuration

is required to enable support for vended credentials and to allow access to the table metadata necessary for Iceberg table operations.

In the `values-overrides.yaml` file, find the section to configure your storage provider under the parents, as shown in the following example:

Configuration of the storage for the Open Catalog

```
catalog:
  storage:
    location: <your-object-store-path>
    type: <your-object-store-type>
    ...
```

To configure it, select the tab for your storage provider, and follow the steps:

Amazon S3

S3-compatible

Azure Storage

Google Cloud Storage

To use the Open Catalog with Amazon S3, do the following:

Configure the access to the storage, as described in [Configure Storage Access](#). Creating a Kubernetes secret may be required.

Configure the Open Catalog in the `values-overrides.yaml` file as follows:

Configuration of the storage for the Open Catalog in Amazon S3

```
catalog:
  storage:
    location: s3://<your-bucket>/<your-folder>
    type: S3
    s3:
      region: <bucket_region>
      roleArn: <dremio_catalog_iam_role> // The role that was configured in the previous
step
      userArn: <dremio_catalog_user_arn> // The IAM user that was created in the
previous step
      externalId: <dremio_catalog_external_id> // The external id that was created in
the previous step
      useAccessKeys: false // Set it to true if you intend to use accessKeys.
    ...
```

If using EKS Pod Identities, ensure the catalog's Kubernetes `ServiceAccount`, which is `dremio-catalog-server` by default, is associated with the `userArn` which you also provided above.

To use the Open Catalog with S3-compatible storage, do the following:

Configure the access to the storage, as described in [Configure Storage Access](#). Creating

a Kubernetes secret is required.

For this step, select the tab for whether the S3-compatible storage has STS support or not, and follow the instructions:

Has STS support

No STS support

The Open Catalog uses STS as a mechanism to perform credentials vending, so configure Open Catalog in the `values-overrides.yaml` file as follows:

caution

roleArn must be provided even when using S3-compatible storage. A dummy value is provided in the template below.

Configuration of the storage for the Open Catalog in S3-compatible with STS support

```
catalog:
  storage:
    location: s3://<your-bucket/<your-folder>
    type: S3
    s3:
      region: <your-bucket-region> // Optional, bucket region
      roleArn: arn:aws:iam::000000000000:role/catalog-access-role // Mandatory, a dummy
role, as shown here, must be provided
      endpoint: <s3-compatible-server-url> // This is the S3 server url, for example,
http://<minio-host>:<minio-port> for MinIO
      stsEndpoint: <s3-compatible-sts-server-url> // This is the STS server url, for
example http://<minio-host>:<minio-port> for MinIO
      pathStyleAccess: true // Mandatory to be true
      useAccessKeys: true // Mandatory to be true
    ...
```

Vended credentials will not work, and, in such cases, you must select `Use master storage credentials` and in the Dremio console, and provide explicit access keys for external engines where they are required.

Once the Kubernetes secrets for the access keys have been created, configure the Open Catalog in the `values-overrides.yaml` file as follows:

caution

roleArn must be provided even when using S3-compatible storage. A dummy value is provided in the template below.

Configuration of the storage for the Open Catalog in S3-compatible with no STS support

```
catalog:
  storage:
    location: s3://<your-bucket/<your-folder>
    type: S3
    s3:
      region: <your-bucket-region> // Optional, bucket region
      roleArn: arn:aws:iam::000000000000:role/catalog-access-role // Mandatory, a dummy
```

```

role, as shown here, must be provided
  endpoint: <s3-compatible-server-url> // This is the S3 server url, for example to
MinIO http://<minio-host>:<minio-port>
  pathStyleAccess: true // Mandatory to be true
  skipSts: true // Mandatory to be true
  useAccessKeys: true // Mandatory to be true
  ...

```

To use the Open Catalog with Azure Storage, do the following:

Configure the access to the storage, as described in [Configure Storage Access](#).

Configure the Open Catalog in the `values-overrides.yaml` file as follows:

Configuration of the storage for the Open Catalog in Azure Storage

```

catalog:
  storage:
                                                                    location:
abfss://<your-container-name>@<your-storage-account>.dfs.core.windows.net/<path>
  type: azure
  azure:
    tenantId: <your-azure-directory-tenant-id>
    multiTenantAppName: ~ // Optional: Used only if you register an app with
multi-tenants.
    useClientSecrets: true // Has to be true
  ...

```

To use the Open Catalog with Google Cloud Storage (GCS), do the following:

Configure the access to the storage, as described in [Configure Storage Access](#).

Configure the Open Catalog in the `values-overrides.yaml` file as follows:

Configuration of the storage for the Open Catalog in Google Cloud Storage

```

catalog:
  ...
  storage:
    location: gs://<your-bucket>/<your-path>
    type: GCS
    gcs:
      useCredentialsFile: True

```

Configuring TLS for Open Catalog External Access

For clients connecting to the Open Catalog from outside the namespace, Transport Layer Security (TLS) can be enabled for Open Catalog external access as follows:

Enable external access with TLS and provide the TLS secret. See the section [Creating a TLS Secret](#).

In the ``values-overrides.yaml`` file, find the Open Catalog configuration section:

Configuration section for the Open Catalog

```
catalog:
  ...
```

Configure TLS for the Open Catalog as follows:

Configuration of TLS for external access to the Open Catalog

```
catalog:
  externalAccess:
    enabled: true
    tls:
      enabled: true
      secret: <dremio-tls-secret-catalog></dremio-tls-secret-catalog>
  ...
```

Configuring Open Catalog When the Coordinator Web is Using TLS

When the Dremio coordinator uses Transport Layer Security (TLS) for Web access (i.e., when ``coordinator.web.tls`` is set to ``true``), the Open Catalog external access must be configured appropriately, or client authentication will fail. For that, configure the Open Catalog as follows:

In the ``values-overrides.yaml`` file, find the Open Catalog configuration section:

Configuration section for the Open Catalog

```
catalog:
  ...
```

Configure the Open Catalog as follows:

Configuration of the Open Catalog when the coordinator web is using TLS

```
catalog:
  externalAccess:
    enabled: true
    authentication:
      authServerHostname: dremio-master-0.dremio-cluster-pod.{{ .Release.Namespace }}.svc.cluster.local
  ...
```

The ``authServerHostname`` must match the CN (or the SAN) field of the (master) coordinator Web TLS certificate.

In case it does not match the CN or SAN fields of the TLS certificate, as a last resort, it is possible to disable hostname verification (``disableHostnameVerification: true``):

Configuration of the Open Catalog with hostname verification disabled

```
catalog:
  externalAccess:
    enabled: true
    authentication:
      authServerHostname: dremio-master-0.dremio-cluster-pod.{{ .Release.Namespace }}.svc.cluster.local
      disableHostnameVerification: true
  ...
```

Downloading Dremio's Helm Charts

You can download Dremio's Helm charts to implement advanced configurations beyond those outlined in this topic.

However, please proceed with caution. Modifications made without a clear understanding can lead to unexpected behavior and compromise the Dremio Support team's ability to provide effective assistance.

To ensure success, Dremio recommends engaging with the Professional Services team through your Account Executive or Customer Success Manager. Please note that such engagements may require additional time and could involve consulting fees.

To download Dremio's Helm charts, use the following command:

Run `helm pull` to download Dremio's Helm charts

```
helm pull oci://quay.io/dremio/dremio-helm --version <tag> --untar
```

Where:

(Optional) ``version <tag>`` - The Helm chart version to pull. For example, ``version 3.0.0``. If not specified, the latest version is pulled.

The command creates a new local directory called ``dremio-helm`` containing the Helm charts.

For more information on the command, see [Helm Pull](#) in Helm's documentation.

Overriding Additional Values

After completing the ``helm pull``:

Find the ``values.yaml`` file, open it, and check the configurations you want to override.

Copy what you want to override from the ``values.yaml`` to ``values-overrides.yaml`` and configure the file with your values.

Save the ``values-overrides.yaml`` file.

Once done with the configuration, deploy Dremio to Kubernetes via the OCI Repo. See how in [Deploying Dremio to Kubernetes](#).

Manual Modifications to Deployment Files

important

For modifications in these files to take effect, you need to install Dremio using a local version of the Helm charts. Thus, the `helm install` command must reference a local folder, not the OCI repo like Quay. For more information and sample commands, see [Helm install](#).

After completing the `helm pull`, you can edit the charts directly. This may be necessary to add deployment-specific modifications not catered for in the Additional Configuration section. These would typically require modifications to files in the `/config` directory. Any customizations to your Dremio environment are propagated to all the pods when installing or upgrading the deployment.

Was this page helpful?

[\[Previous](#)

[Deploy on Kubernetes\]\(/current/deploy-dremio/deploy-on-kubernetes\)\[Next](#)

[Managing Engines\]\(/current/deploy-dremio/managing-engines-kubernetes\)](#)

[Configure Your Values](#)

[License](#)

[Pull Secret](#)

[Coordinator](#)

[Resource Configuration](#)

[Identity Provider](#)

[Transport Level Security](#)

[Coordinator's Distributed Storage](#)

[Open Catalog](#)

[Configuring Your Values - Advanced](#)

[OpenShift](#)

[Dremio Platform Images](#)

[Scale-out Coordinators](#)

[Configuring Kubernetes Pod Metadata \(including Node Selector\)](#)

[Configuring Pods Priority](#)

[Configuring Extra Environment Variables](#)

[Advanced Load Balancer Configuration](#)

[Additional Load Balancer Configuration for Amazon EKS in Auto Mode](#)

Advanced TLS Configuration for OpenSearch

Advanced Configuration of Engines

Configuration of Classic Engines

Engine Overrides

Telemetry

Logging

Disabling Parts of the Deployment

Semantic Search

Additional Configuration

Additional Config Files

Additional Config Variables

Additional Java Truststore

Additional Config Binary Files

Hive

References

Recommended Resources Configuration

Creating a TLS Secret

Configuring the Distributed Storage

Configuring Storage for the Open Catalog

Configuring TLS for Open Catalog External Access

Configuring Open Catalog When the Coordinator Web is Using TLS

Downloading Dremio's Helm Charts

Overriding Additional Values

Manual Modifications to Deployment Files

Source:
<https://docs.dremio.com/current/deploy-dremio/managing-engines-kubernetes>

Version: current [26.x]

On this page

Managing Engines in Kubernetes Enterprise

note

This feature is for Enterprise Edition only.
For Community Edition, see [Configuration of Classic Engines](#).

Dremio supports the ability to provision multiple separate execution engines in Kubernetes from a Dremio main coordinator node, and automatically start and stop based on workload requirements at runtime. This provides several benefits, including:

Creating a new engine doesn't require restarting Dremio, which enables administrators to achieve workload isolation efficiently.

When creating a new engine, you can use Kubernetes metadata to label engines to keep track of resources.

Right-size execution resources for each distinct workload, instead of implementing a one-size-fits-all model.

Easily experiment with different execution resource sizes at any scale.

To manage your engines, open the Engines page as follows:

Open your Dremio console.

Click [!The Settings icon](#) in the side navigation bar to open the Settings sidebar.

Select **Engines**.

[!Engines page under Project Settings listing all engines.](#)

Monitoring Engines

You can monitor the status and properties of your engines on the Engines page.

[!Engines page under Project Settings showing all the current engines and their statuses.](#)

Each engine has the following information available:

Name - The name of the engine, which you can click to see its details. See the section about Viewing Engine Details.

Size - The size configured for the engine.

Status - The engine status. For more information, see the section in this topic about Engine Statuses.

Auto start/stop - Whether the engine has auto start/stop enabled for autoscaling.

Idle period - The idle time to auto stop when the engine has **Auto start/stop** enabled.

Queues - Query queues routed to the engine.

Labels - Labels associated with the engine.

Performing Actions on Engines

While monitoring engines, you have actions you can perform on each engine through the icons displayed on the right-hand side when hovering over the engine row.

[!Engines page showing the icons with the actions for each engine.](#)

Stopping/Starting an Engine

You can click [!The stop engine icon](#)/[!The start engine icon](#) to stop/start an engine manually at any time. Stopping an engine will cause running queries to fail while new queries will remain queued, which can also fail by timeout if the engine isn't started. To prevent query failures, reroute queries to another engine, and stop the engine only when no queries are running or queued for the engine.

note

You can enable **autoscaling on an engine** to make it stop automatically after an idle time without queries and start again automatically when new queries are issued, all without any human intervention.

Autoscaling is configured when you add an engine or edit an engine:

Stopping All Engines

Some complex operations, like upgrading or uninstalling Dremio, require all engines to be stopped beforehand. You can stop engines manually one by one as described above, or automate the procedure using the [Engine Management API](#) to stop all engines. Expand the sample below of a bash script executing the necessary endpoints to stop all engines.

Sample bash script to stop all engines

```
#!/bin/bash
# Check if the bearer token is provided
if [ -z "$1" ]; then
echo "Error: Bearer token is required."
exit 1
fi
BEARER_TOKEN=$1
BASE_URL=${2:-https://localhost:9047}
# Make an HTTP GET request to retrieve engine IDs
RESPONSE=$(curl -k -s -H "Authorization: Bearer $BEARER_TOKEN"
"$BASE_URL/api/v3/engines")
# Check if the response contains the "id" field
if ! echo "$RESPONSE" | grep -q '"id"'; then
echo "Error: No 'id' field found in the response."
exit 1
fi
# Extract IDs from the response
IDS=$(echo "$RESPONSE" | jq -r '.data[] | .id')
# Loop through each ID and make an HTTP PUT request
for ID in $IDS; do
RESPONSE=$(curl -k -s -o /dev/null -w "%{http_code}" -X PUT -H "Authorization: Bearer
$BEARER_TOKEN" "$BASE_URL/api/v3/engines/$ID/stop")
if [ "$RESPONSE" -eq 200 ]; then
```

```
    echo "Successfully stopped engine with ID: $ID"
else
    echo "Failed to stop the engine with ID: $ID, HTTP status code: $RESPONSE"
fi
done
echo "All engines processed."
```

Editing the Engine Settings

You can click [!The edit engine icon](#) to edit the engine settings. After saving the new settings, the engine may restart, causing running queries to fail and new queries to be queued.

[!Edit engine showing the general settings.](#)

note

The name of the engine must follow these rules:

Must start with a lowercase alphanumeric character (``[a-z0-9]``).

Must end with a lowercase alphanumeric character (``[a-z0-9]``).

Must contain only lowercase alphanumeric characters or a hyphen (``[\-a-z0-9]``).

Must be under 30 characters in length.

Must be unique and not previously used for any existing or deleted engines.

Deleting an Engine

You can click [!The delete engine icon](#) to delete an engine. Deleting an engine will cause running, queued, and new queries to fail. To prevent query failures, you can reroute queries to another engine, and only delete when no more queries are running or queued for the engine.

Viewing Engine Details

While monitoring engines, if you need to know more details about engines, click the engine's name to view all the information about it.

[!Engine details page showing all the information about the engine.](#)

On this page, you will also find a set of buttons at the top to delete the engine, stop/start the engine, and edit the engine settings.

Adding an Engine

You can create more engines by clicking **Add Engine** at the top-right corner of the Engines page.

!The general settings to add a new engine.

In the New engine dialog, do the following:

Fill out the **General** section:

Name - Type the name of the engine. Use a meaningful name that helps you to identify the engine better. For example, `low-cost-query`.

note

The name of the engine must follow these rules:

Must start with a lowercase alphanumeric character (`[a-z0-9`).

Must end with a lowercase alphanumeric character (`[a-z0-9`).

Must contain only lowercase alphanumeric characters or a hyphen (`[-a-z0-9`).

Must be under 30 characters in length.

Must be unique and not previously used for any existing or deleted engines.

CPU and **Size** - Select the number of CPUs per executor pod and the size of the engine. Dremio provides nine engine sizes, each with two CPU options targeting 16 or 32 CPU nodes. By default, Dremio will subtract 2 CPUs and 8 GB of memory from its request, resulting in requests for 14 or 30 CPUs and 120 GB of memory. This adjustment helps optimize the packing of executors on the most common node sizes. The table below shows the engine sizes:

Engine Size	Executors per Replica	Memory per Executor
---	---	---
2XSmall	1	56 GB
XSmall	1	120 GB
Small	2	120 GB
Medium	4	120 GB
Large	8	120 GB
XLarge	12	120 GB
2XLarge	16	120 GB
3XLarge	24	120 GB
4XLarge	32	120 GB

Automatically start/stop - If checked, the engine automatically stops after the specified idle time and automatically starts when new queries are issued to the engine. If not checked, the engine only stops and starts through manual intervention. By default, this setting is checked and the engine stops automatically after `15 min` of idle time. For more information, see the section Stopping/Starting an Engine.

(Optional) Expand **Advanced Options** for further settings.

!The advanced options to add a new engine.

Fill out the advanced options as follows:

Cloud cache volume (c3) - Specify the amount of local storage for caching data.

Spill volume - Specify the disk size allocated for temporary storage when operations exceed memory limits.

(Optional) Select **Kubernetes pod metadata** to define pod metadata for the engine, such as labels, annotations, node selectors, and tolerations. Define those values with care and foreknowledge of expected entries because any misconfiguration may result in Kubernetes being unable to start the executors that make up the engine.

!The metadata to add an engine.

Fill out the pod's metadata with:

Labels - Add labels as key/value pairs to identify and organize pods. Use them to group, filter, and select subsets of resources efficiently.

note

The engine label must follow these rules:

Must start with an alphanumeric character ([a-z0-9]).

Must end with an alphanumeric character ([a-z0-9]).

Must contain only lowercase alphanumeric characters, a hyphen, or an underscore ([a-z0-9_-]).

The maximum length is 63 characters.

Annotations - Add annotations as key/value pairs to store non-identifying metadata, such as build information or pointers to logging services. Unlike labels, they're not used for selection or grouping.

note

The engine annotation must follow these rules:

Must be UTF-8 encoded and can include any valid UTF-8 character.

Can be in plain text, JSON, or any other UTF-8 compatible format.

The maximum size is 256KB.

The maximum size of all engine annotations is 1MB.

Node selectors - Add node selectors as key/value pairs for node-specific constraints to schedule pods on nodes matching specified labels. Use this to target nodes with specific configurations or roles.

Tolerations - Add tolerations to allow pods to be scheduled on nodes with matching taints, but they don't restrict scheduling to only those nodes; the pod can still land on a node without the taint.

Click **Add** to add the engine.

The newly added engine will be displayed in the listed engines.

!Engines page showing the current engines, which now includes the newly added engine.

Engine Statues

The following table describes each engine status:

Status	Icon	Description
---	---	---
Starting	The starting engine icon	The engine is starting. This is the initial state of an engine after being created. New queries are queued to be processed.
Running	The running engine icon	The engine is running. New queries are queued and processed.
Stopping	The stopping engine icon	The engine is stopping. Running queries will fail. New queries will remain queued, which can also fail by timeout if the engine isn't started.
Stopped	The stopped engine icon	The engine is stopped. New queries will remain queued, which can fail by timeout if the engine isn't started.
Recovering	The recovering engine icon	The engine is recovering. New queries will remain queued, which can fail by timeout if the engine doesn't recover.
Failed	The failed engine icon	The engine failed. New queries will remain queued, which can fail by timeout if the engine doesn't start.

Related Topics

[Engine Management API](#) - The API to manage your engines using REST API calls.

[sys.engines](#) - The system table to query for information about your engines.

[Audit Logs](#) - Audit logs for your engines.

Was this page helpful?

[\[Previous](#)

[Configuring Your Values\]\(/current/deploy-dremio/configuring-kubernetes/\)](#)[\[Next](#)

[Other Options\]\(/current/deploy-dremio/other-options/\)](#)

[Monitoring Engines](#)

[Performing Actions on Engines](#)

[Stopping/Starting an Engine](#)

[Editing the Engine Settings](#)

[Deleting an Engine](#)

[Viewing Engine Details](#)

[Adding an Engine](#)

[Engine Statuses](#)

[Related Topics](#)

Source:
<https://docs.dremio.com/current/deploy-dremio/other-options/>

Version: current [26.x]

Other Deployment Options

Besides the [Kubernetes deployment](#), there are other alternative supported options for deploying Dremio:

[Hadoop Deployment \(YARN\)](#) - Deploy Dremio on a Hadoop cluster using YARN.

[Dremio on Your Infrastructure](#) - Deploy Dremio as a standalone cluster.

Was this page helpful?

[\[Previous](#)

[Managing Engines\]\(/current/deploy-dremio/managing-engines-kubernetes\)](#)[\[Next](#)

[Dremio with Hadoop\]\(/current/deploy-dremio/other-options/yarn-hadoop\)](#)

Source:
<https://docs.dremio.com/current/deploy-dremio/kubernetes-environments/>

Version: current [26.x]

On this page

Kubernetes Environments for Dremio

Dremio is designed to run Kubernetes environments, providing enterprise-grade data lakehouse capabilities. To successfully [deploy Dremio on Kubernetes](#), you need a compatible hosted Kubernetes environment.

Dremio is tested and supported on the following Kubernetes environments:

Elastic Kubernetes Service (EKS)

Azure Kubernetes Service (AKS)

Google Kubernetes Engine (GKE)

Red Hat OpenShift

The sections on this page detail recommendations for AWS and Azure. Please use the information provided as a guide for your vendors' equivalent options.

note

If you're using a containerization platform built on Kubernetes that isn't listed here, please contact your provider and Dremio Account team to discuss compatibility and support options.

Requirements

Versions

Dremio requires regular updates to your Kubernetes version. You must be on an officially supported version, and preferably not one on extended support. See the following examples for AWS [Available versions on standard support](#) and Azure [Kubernetes versions](#).

Recommendations

See this table for resource request recommendations of the various parts of the deployment, [Recommended Resources Configuration](#).

For a list of all Dremio engine sizes see, [Add an Engine](#). Engines will make up the lions share of any Dremio deployment.

Node Sizes

The following sections suggest AWS and Azure machines that could be used to meet our recommendations.

Dremio recommends having separate EKS node groups for the different components of our services to allow each node group to autoscale independently:

Core Services

Coordinators

For [coordinators](#), Dremio recommends at least 32 CPUs and 64 GB of memory, hence, a ``c6i.8xlarge`` or ``Standard_F32s_v2`` is a good option, offering a CPU-to-memory ratio of 1:2. In the Helm charts, this would result in 30 CPUs and 60 GB of memory allocated to the Dremio pod.

Executors

For [executors](#), Dremio recommends either:

16 CPUs and 128 GB of memory, hence, a ``r5d.4xlarge`` or ``Standard_E16_v5`` is a good option, offering a CPU-to-memory ratio of 1:8. In the Helm charts, this results in 15 CPUs and 120 GB of memory allocated to the Dremio pod.

32 CPUs and 128 GB of memory, hence, a ``m5d.8xlarge`` or ``Standard_D32_v5`` is a good option, offering a CPU-to-memory ratio of 1:4 for high-concurrency workloads. In the Helm charts, this results in 30 CPUs and 120 GB of memory allocated to the Dremio pod.

Auxiliary Services

[Open Catalog](#) and [Semantic Search](#).

Catalog is made up of 4 key components: Catalog Service, Catalog Server, Catalog External, and MongoDB. Search has one key component, OpenSearch.

Each of these components needs between 2-4 CPUs and 4-16 GB of memory; hence, a ``m5d.2xlarge`` or ``Standard_D8_v5`` is a good option and could be used to host multiple containers that are part of these services.

ZooKeeper, NATS, Operators, and Open Telemetry:

Each of these need between 0.5-1 CPUs and 0.5-1 GB, `m5d.large`, `t2.medium`, `Standard_D2_v5` or `Standard_A2_v2` are good options and could be used to host multiple containers that are part of these services.

Disk Storage Class

Dremio recommends:

For AWS, GP3 or IO2 as the storage type for all nodes.

For Azure managed-premium as the storage type for all nodes.

Additionally, for coordinators and executors, you can further use local NVMe SSD storage for C3 and spill on executors. For more information on storage classes, see the following resources [AWS Storage Class](#) and [Azure Storage Class](#).

Storage size requirements are:

Coordinator volume #1: 128-512 GB (key-value store).

Coordinator volume #2: 16 GB (logs).

Executor volume #1: 128-512 GB (spilling).

Executor volume #2: 128-512 GB (C3).

Executor volume #3: 16 GB (logs).

MongoDB volume: 128-512 GB.

OpenSearch volume: 128 GB.

Zookeeper volume: 16 GB.

EKS Add-Ons

The following add-ons are required for EKS clusters:

Amazon EBS CSI Driver

EKS Pod Identity Agent

Was this page helpful?

[\[Previous](#)

[Deploy Dremio\]\(/current/deploy-dremio/\)\[Next](#)

[Deploy on Kubernetes\]\(/current/deploy-dremio/deploy-on-kubernetes\)](#)

[Requirements](#)

[Versions](#)

-

Source:
<https://docs.dremio.com/current/deploy-dremio/deploy-on-kubernetes/>

Version: current [26.x]

On this page

Deploy Dremio on Kubernetes

You can follow these instructions to deploy Dremio on Kubernetes provisioned through a cloud provider or running in an on-premises environment.

FREE TRIAL

If you are using an **Enterprise Edition free trial**, go to [Get Started with the Enterprise Edition Free Trial](#).

Prerequisites

Before deploying Dremio on Kubernetes, ensure you have the following:

A hosted Kubernetes environment to deploy and manage the Dremio cluster. Each Dremio release is tested against [Amazon Elastic Kubernetes Service \(EKS\)](#), [Azure Kubernetes Service \(AKS\)](#), and [Google Kubernetes Engines \(GKE\)](#) to ensure compatibility. If you have a containerization platform built on top of Kubernetes that is not listed here, please contact your provider and the Dremio Account Team regarding compatibility.

Helm 3 installed on your local machine to run Helm commands. For installation instructions, refer to [Installing Helm](#) in the Helm documentation.

A local kubectl configured to access your Kubernetes cluster. For installation instructions, refer to [kubectl](#) in the Kubernetes documentation.

Object Storage: Amazon S3 (including S3-compatible, e.g., MinIO), Azure Storage, or Google Cloud Storage (GCS).

Storage classes that support ReadWriteOnce (RWO) access mode and ideally can create expandable volumes.

The ability to connect to [Quay.io](#) to access the [new v3 Helm chart](#) for Dremio 26+, since the [older v2 Helm chart](#) will not function.

Additional Prerequisites for the Enterprise Edition

For the Enterprise Edition, you must:

Create an account on [Quay.io](#) to access [Dremio's OCI repository](#), which stores Dremio's

Helm charts and images.

To get access, contact your Dremio account executive or Dremio Support.

note

If your internet access doesn't allow reaching Dremio's OCI repository in Quay.io, consider using a private mirror to fetch Dremio's Helm chart images.

Get a valid license key issued by Dremio to put in the Helm chart. To obtain the license, refer to [Licensing](#).

Additional Prerequisites for the OpenShift

Before deploying Dremio onto OpenShift, you additionally need the following:

Have the OpenShift `oc` CLI command configured and authenticated. For the installation instructions, see [OpenShift CLI \(oc\)](#).

Node Tuning for OpenSearch on OpenShift

OpenSearch requires the `vm.max_map_count` kernel parameter to be set to at least **262144**.

This parameter controls the maximum number of memory map areas a process can have, and OpenSearch uses memory-mapped files extensively for performance.

Without this setting, OpenSearch pods will fail to start with errors related to virtual memory limits.

Since the Helm chart sets `setVMMMaxMapCount: false` for OpenShift compatibility (to avoid privileged init containers), you need to configure this kernel parameter at the node level. The **recommended way** to do it is a Node Tuning Operator. This Operator ships with OpenShift and provides a declarative way to configure kernel parameters.

Create a `Tuned` resource to configure the required kernel parameter:

The `tuned-opensearch.yaml` configuration file

```
apiVersion: tuned.openshift.io/v1
kind: Tuned
metadata:
  name: opensearch-opensearch
  namespace: opensearch-cluster-node-tuning-operator
spec:
  profile:
    - data: |
        [main]
        summary=Optimize systems running OpenSearch on OpenShift nodes
        include=opensearch-node
        [sysctl]
        vm.max_map_count=262144
        name: opensearch-opensearch
  recommend:
    - match:
        - label: tuned.openshift.io/opensearch
```

```
type: pod
priority: 20
profile: openshift-opensearch
```

This YAML should be saved locally and applied to any cluster you intend to deploy Dremio:

```
oc apply -f tuned-opensearch.yaml
```

Step 1: Deploy Dremio

To deploy the Dremio cluster in Kubernetes, do the following:

Configure your values to deploy Dremio to Kubernetes in the file ``values-overrides.yaml``. For that, go to [Configuring Your Values to Deploy Dremio to Kubernetes](#) and get back here to continue with the deployment.

On your terminal, start the deployment by installing Dremio's Helm chart:

Standard Kubernetes

OpenShift

Run the following command for any Kubernetes environment except for OpenShift:

```
helm install <your-dremio-install-release> oci://quay.io/dremio/dremio-helm \
--values <your-local-path>/values-overrides.yaml \
--version <optional-helm-chart-version> \
--set-file <optional-config-files> \
--wait
```

Where:

``<your-dremio-install-release>`` - The name that identifies your Dremio installation. For example, ``dremio-1-0``.

``<your-local-path>`` - The path to reach your ``values-overrides.yaml`` configuration file.

(Optional) ``version <optional-helm-chart-version>`` - The version of Dremio's Helm chart to be used. If not provided, defaults to the latest.

(Optional) ``set-file <optional-config-file>`` - An optional configuration file for deploying Dremio. For example, an Identity Provider configuration file, which is not defined in the ``values-overrides.yaml`` and can be provided here through this option.

For OpenShift, the command requires an additional ``values`` option with the path to the OpenShift-specific ``values-openshift-overrides.yaml`` configuration file. This additional option must be placed before the ``values`` option with the ``values-overrides.yaml`` configuration file, resulting in its substitution first.

Run the following command for OpenShift:

```
helm install <your-dremio-install-release> oci://quay.io/dremio/dremio-helm \
```

```
--values <your-local-path1>/values-openshift-overrides.yaml \  
--values <your-local-path2>/values-overrides.yaml \  
--version <optional-helm-chart-version> \  
--set-file <optional-config-files> \  
--wait
```

Where:

`<your-dremio-install-release>` - The name that identifies your Dremio installation. For example, `dremio-1-0`.

`<your-local-path1>` - The path to reach your `values-openshift-overrides.yaml` configuration file. Only required for OpenShift.

`<your-local-path2>` - The path to reach your `values-overrides.yaml` configuration file.

(Optional) `version <optional-helm-chart-version>` - The version of Dremio's Helm chart to be used. If not provided, defaults to the latest.

(Optional) `set-file <optional-config-file>` - An optional configuration file for deploying Dremio. For example, an Identity Provider configuration file, which is not defined in the `values-overrides.yaml` and can be provided here through this option.

Monitor the deployment using the following commands:

Standard Kubernetes

OpenShift

Run the following command for any Kubernetes environment except for OpenShift:

```
kubectl get pods
```

For OpenShift, run the following command:

```
oc get pods
```

When all of the pods are in the `Ready` state, the deployment is complete.

Troubleshooting

If a pod remains in `Pending` state for more than a few minutes, run the following command to view its status to check for issues, such as insufficient resources for scheduling:

```
kubectl describe pods <pod-name>
```

If the events at the bottom of the output mention insufficient CPU or memory, do one of the following:

Adjust the values in the `values-overrides.yaml` configuration file and redeploy.

Add more resources to your Kubernetes cluster.

If a pod returns a failed state (especially `dremio-master-0`, the most important pod), use the following commands to collect the logs:

Standard Kubernetes

OpenShift

Run the following command for any Kubernetes environment except for OpenShift:

```
kubectll logs dremio-master-0
```

For OpenShift, run the following command:

```
oc logs deployment/dremio-master
```

Step 2: Connecting to Dremio

Now that you've installed the Helm chart and deployed Dremio on Kubernetes, the next step is connecting to Dremio, where you have the following options:

Dremio Console

OpenShift Route

BI Tools via ODBC/JDBC

BI Tools via Apache Arrow Flight

To connect to Dremio via [the Dremio console](#), run the following command to use the `services dremio-client` in Kubernetes to find the host for the Dremio console:

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
AGE				
...
...	...			

If the value in the `TYPE` column of the output is `LoadBalancer`, access the Dremio console through the address in the `EXTERNAL_IP` column and port **9047**.

For example, in the output below, the value under the `EXTERNAL-IP` column is `8.8.8.8`. Therefore, access the Dremio console through `<http://8.8.8.8:9047>`.

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
AGE				
dremio-client	LoadBalancer	10.99.227.180	8.8.8.8	
31010:32260/TCP,9047:30620/TCP	2d			

If you want to change the exposed port on the load balancer, change the value of the

setting `coordinator.web.port` in the file `values-overrides.yaml`.

If the value in the `TYPE` column of the output is `NodePort`, access the Dremio console through `<http://localhost:30670>`.

To expose Dremio externally using OpenShift Routes, do the following:

```
$ oc expose service dremio-client --port=9047 --name=dremio-ui

$ oc get route dremio-ui -o jsonpath='{.spec.host}'
```

To connect your BI tools to Dremio via ODBC/JDBC, run the following command to use the `services dremio-client` in Kubernetes to find the host for ODBC/JDBC connections by using the following command:

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
AGE				
...
...	...			

If the value in the `TYPE` column of the output is `LoadBalancer`, access Dremio using ODBC/JDBC through the address in the `EXTERNAL_IP` column and port **31010**.

For example, in the output below, the value under the `EXTERNAL-IP` column is `8.8.8.8`. Therefore, access Dremio using ODBC/JDBC on port 31010 through `<http://8.8.8.8:31010>`.

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
AGE				
dremio-client	LoadBalancer	10.99.227.180	8.8.8.8	31010:32260/TCP,9047:30620/TCP
		2d		

If you want to change the exposed port on the load balancer, change the value of the setting `coordinator.client.port` in the file `values-overrides.yaml`.

If the value in the `TYPE` column of the output is `NodePort`, access Dremio using ODBC/JDBC through `<http://localhost:32390>`.

To connect your BI tools to Dremio via Apache Arrow Flight, run the following command to use the `services dremio-client` in Kubernetes to find the host for Apache Arrow Flight connections by using the following command:

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
AGE				
...
...	...			

If the value in the `TYPE` column of the output is `LoadBalancer`, access Dremio using Apache Arrow Flight through the address in the `EXTERNAL_IP` column and port **32010**.

For example, in the output below, the value under the `EXTERNAL-IP` column is `8.8.8.8`. Therefore, access Dremio using Apache Arrow Flight through `<http://8.8.8.8:32010>`.

```
$ kubectl get services dremio-client
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
	AGE			
dremio-client	LoadBalancer	10.99.227.180	8.8.8.8	31010:32260/TCP,9047:30620/TCP
	2d			

If you want to change the exposed port on the load balancer, change the value of the setting `coordinator.flight.port` in the file `values-overrides.yaml`.

If the value in the `TYPE` column of the output is `NodePort`, access Dremio using Apache Arrow Flight through `<http://localhost:31357>`.

Was this page helpful?

[Previous

Kubernetes Environments](/current/deploy-dremio/kubernetes-environments)[Next

Configuring Your Values](/current/deploy-dremio/configuring-kubernetes/)

Prerequisites

Additional Prerequisites for the Enterprise Edition

Additional Prerequisites for the OpenShift

Step 1: Deploy Dremio

Step 2: Connecting to Dremio

Source:
<https://docs.dremio.com/current/deploy-dremio/managing-engines-kubernetes/>

Version: current [26.x]

On this page

Managing Engines in Kubernetes Enterprise

note

This feature is for Enterprise Edition only.
For Community Edition, see [Configuration of Classic Engines](#).

Dremio supports the ability to provision multiple separate execution engines in Kubernetes from a Dremio main coordinator node, and automatically start and stop based on workload requirements at runtime. This provides several benefits, including:

Creating a new engine doesn't require restarting Dremio, which enables administrators

to achieve workload isolation efficiently.

When creating a new engine, you can use Kubernetes metadata to label engines to keep track of resources.

Right-size execution resources for each distinct workload, instead of implementing a one-size-fits-all model.

Easily experiment with different execution resource sizes at any scale.

To manage your engines, open the Engines page as follows:

Open your Dremio console.

Click [!The Settings icon](#) in the side navigation bar to open the Settings sidebar.

Select **Engines**.

[!Engines page under Project Settings listing all engines.](#)

Monitoring Engines

You can monitor the status and properties of your engines on the Engines page.

[!Engines page under Project Settings showing all the current engines and their statuses.](#)

Each engine has the following information available:

Name - The name of the engine, which you can click to see its details. See the section about Viewing Engine Details.

Size - The size configured for the engine.

Status - The engine status. For more information, see the section in this topic about Engine Statuses.

Auto start/stop - Whether the engine has auto start/stop enabled for autoscaling.

Idle period - The idle time to auto stop when the engine has **Auto start/stop** enabled.

Queues - Query queues routed to the engine.

Labels - Labels associated with the engine.

Performing Actions on Engines

While monitoring engines, you have actions you can perform on each engine through the icons displayed on the right-hand side when hovering over the engine row.

[!Engines page showing the icons with the actions for each engine.](#)

Stopping/Starting an Engine

You can click [!The stop engine icon](#)/[!The start engine icon](#) to stop/start an engine

manually at any time. Stopping an engine will cause running queries to fail while new queries will remain queued, which can also fail by timeout if the engine isn't started. To prevent query failures, reroute queries to another engine, and stop the engine only when no queries are running or queued for the engine.

note

You can enable **autoscaling on an engine** to make it stop automatically after an idle time without queries and start again automatically when new queries are issued, all without any human intervention.

Autoscaling is configured when you add an engine or edit an engine:

Stopping All Engines

Some complex operations, like upgrading or uninstalling Dremio, require all engines to be stopped beforehand. You can stop engines manually one by one as described above, or automate the procedure using the [Engine Management API](#) to stop all engines. Expand the sample below of a bash script executing the necessary endpoints to stop all engines.

Sample bash script to stop all engines

```
#!/bin/bash
# Check if the bearer token is provided
if [ -z "$1" ]; then
echo "Error: Bearer token is required."
exit 1
fi
BEARER_TOKEN=$1
BASE_URL=${2:-https://localhost:9047}
# Make an HTTP GET request to retrieve engine IDs
RESPONSE=$(curl -k -s -H "Authorization: Bearer $BEARER_TOKEN"
"$BASE_URL/api/v3/engines")
# Check if the response contains the "id" field
if ! echo "$RESPONSE" | grep -q '"id"'; then
echo "Error: No 'id' field found in the response."
exit 1
fi
# Extract IDs from the response
IDS=$(echo "$RESPONSE" | jq -r '.data[] | .id')
# Loop through each ID and make an HTTP PUT request
for ID in $IDS; do
RESPONSE=$(curl -k -s -o /dev/null -w "%{http_code}" -X PUT -H "Authorization: Bearer
$BEARER_TOKEN" "$BASE_URL/api/v3/engines/$ID/stop")
if [ "$RESPONSE" -eq 200 ]; then
echo "Successfully stopped engine with ID: $ID"
else
echo "Failed to stop the engine with ID: $ID, HTTP status code: $RESPONSE"
fi
done
echo "All engines processed."
```

Editing the Engine Settings

You can click [!The edit engine icon](#) to edit the engine settings. After saving the new settings, the engine may restart, causing running queries to fail and new queries to be queued.

[!Edit engine showing the general settings.](#)

note

The name of the engine must follow these rules:

Must start with a lowercase alphanumeric character (`[a-z0-9]`).

Must end with a lowercase alphanumeric character (`[a-z0-9]`).

Must contain only lowercase alphanumeric characters or a hyphen (`[\-a-z0-9]`).

Must be under 30 characters in length.

Must be unique and not previously used for any existing or deleted engines.

Deleting an Engine

You can click [!The delete engine icon](#) to delete an engine. Deleting an engine will cause running, queued, and new queries to fail. To prevent query failures, you can reroute queries to another engine, and only delete when no more queries are running or queued for the engine.

Viewing Engine Details

While monitoring engines, if you need to know more details about engines, click the engine's name to view all the information about it.

[!Engine details page showing all the information about the engine.](#)

On this page, you will also find a set of buttons at the top to delete the engine, stop/start the engine, and edit the engine settings.

Adding an Engine

You can create more engines by clicking **Add Engine** at the top-right corner of the Engines page.

[!The general settings to add a new engine.](#)

In the New engine dialog, do the following:

Fill out the **General** section:

Name - Type the name of the engine. Use a meaningful name that helps you to identify the engine better. For example, `low-cost-query`.

note

The name of the engine must follow these rules:

Must start with a lowercase alphanumeric character ([a-z0-9]).

Must end with a lowercase alphanumeric character ([a-z0-9]).

Must contain only lowercase alphanumeric characters or a hyphen ([\a-z0-9]).

Must be under 30 characters in length.

Must be unique and not previously used for any existing or deleted engines.

CPU and Size - Select the number of CPUs per executor pod and the size of the engine. Dremio provides nine engine sizes, each with two CPU options targeting 16 or 32 CPU nodes. By default, Dremio will subtract 2 CPUs and 8 GB of memory from its request, resulting in requests for 14 or 30 CPUs and 120 GB of memory. This adjustment helps optimize the packing of executors on the most common node sizes. The table below shows the engine sizes:

Engine Size	Executors per Replica	Memory per Executor
---	---	---
2XSmall	1	56 GB
XSmall	1	120 GB
Small	2	120 GB
Medium	4	120 GB
Large	8	120 GB
XLarge	12	120 GB
2XLarge	16	120 GB
3XLarge	24	120 GB
4XLarge	32	120 GB

Automatically start/stop - If checked, the engine automatically stops after the specified idle time and automatically starts when new queries are issued to the engine. If not checked, the engine only stops and starts through manual intervention. By default, this setting is checked and the engine stops automatically after `15 min` of idle time. For more information, see the section Stopping/Starting an Engine.

(Optional) Expand **Advanced Options** for further settings.

!The advanced options to add a new engine.

Fill out the advanced options as follows:

Cloud cache volume (c3) - Specify the amount of local storage for caching data.

Spill volume - Specify the disk size allocated for temporary storage when operations exceed memory limits.

(Optional) Select **Kubernetes pod metadata** to define pod metadata for the engine, such as labels, annotations, node selectors, and tolerations. Define those values with care and foreknowledge of expected entries because any misconfiguration may result in Kubernetes being unable to start the executors that make up the engine.

!The metadata to add an engine.

Fill out the pod's metadata with:

-

Labels - Add labels as key/value pairs to identify and organize pods. Use them to group, filter, and select subsets of resources efficiently.

note

The engine label must follow these rules:

Must start with an alphanumeric character ([a-z0-9]).

Must end with an alphanumeric character ([a-z0-9]).

Must contain only lowercase alphanumeric characters, a hyphen, or a underscore ([-_a-z0-9]).

The maximum length is 63 characters.

Annotations - Add annotations as key/value pairs to store non-identifying metadata, such as build information or pointers to logging services. Unlike labels, they're not used for selection or grouping.

note

The engine annotation must follow these rules:

Must be UTF-8 encoded and can include any valid UTF-8 character.

Can be in plain text, JSON, or any other UTF-8 compatible format.

The maximum size is 256KB.

The maximum size of all engine annotations is 1MB.

Node selectors - Add node selectors as key/value pairs for node-specific constraints to schedule pods on nodes matching specified labels. Use this to target nodes with specific configurations or roles.

Tolerations - Add tolerations to allow pods to be scheduled on nodes with matching taints, but they don't restrict scheduling to only those nodes; the pod can still land on a node without the taint.

Click **Add** to add the engine.

The newly added engine will be displayed in the listed engines.

[!Engines page showing the current engines, which now includes the newly added engine.](#)

Engine Statuses

The following table describes each engine status:

Status	Icon	Description
---	---	---
Starting	The starting engine icon	The engine is starting. This is the initial state of an engine after being created. New queries are queued to be processed.
Running	The running engine icon	The engine is running. New queries are queued and processed.

Stopping	The stopping engine icon	The engine is stopping. Running queries will fail. New queries will remain queued, which can also fail by timeout if the engine isn't started.
Stopped	The stopped engine icon	The engine is stopped. New queries will remain queued, which can fail by timeout if the engine isn't started.
Recovering	The recovering engine icon	The engine is recovering. New queries will remain queued, which can fail by timeout if the engine doesn't recover.
Failed	The failed engine icon	The engine failed. New queries will remain queued, which can fail by timeout if the engine doesn't start.

Related Topics

[Engine Management API](#) - The API to manage your engines using REST API calls.

[sys.engines](#) - The system table to query for information about your engines.

[Audit Logs](#) - Audit logs for your engines.

Was this page helpful?

[\[Previous](#)

[Configuring Your Values\]\(/current/deploy-dremio/configuring-kubernetes/\)\[Next](#)

[Other Options\]\(/current/deploy-dremio/other-options/\)](#)

[Monitoring Engines](#)

[Performing Actions on Engines](#)

[Stopping/Starting an Engine](#)

[Editing the Engine Settings](#)

[Deleting an Engine](#)

[Viewing Engine Details](#)

[Adding an Engine](#)

[Engine Statuses](#)

[Related Topics](#)

Source: dremio-software-get-started.md

Dremio Software - Get Started

Source:
<https://docs.dremio.com/current/get-started/>

Version: current [26.x]

On this page

Get Started with Dremio

Welcome to Dremio, a data lakehouse platform that facilitates high-performance, self-service analytics on large datasets.

To get started with Dremio, you have two options for a hands-on experience:

[Enterprise Edition Free Trial](#) (Recommended) - Deploy and explore Dremio on Kubernetes using the Enterprise Edition free trial with all features unlocked.

[Community Edition on Docker](#) - Deploy and explore Dremio on Docker using the Community Edition with a limited set of features. This option provides a local deployment with a single node, which is suggested for testing and evaluation purposes.

Choose the option that best fits your needs.

To learn about the differences between the two editions, see [Dremio Editions](#).

Related Topics

If you want to learn more about Dremio, see the following:

[Quick Tour of the Console](#) - A walkthrough of the Dremio console and how to best use its various capabilities.

[What is Dremio?](#) - An overview of Dremio, its key concepts, and architecture.

Was this page helpful?

[\[Previous](#)

[Overview\]](#)(/current/)[[Next](#)

[Enterprise Edition Free Trial\]](#)(/current/get-started/kubernetes-trial)

[Related Topics](#)

Source:
<https://docs.dremio.com/current/get-started/kubernetes-trial>

Version: current [26.x]

On this page

Get Started with the Enterprise Edition Free Trial

This Get Started guide walks you through deploying Dremio on Kubernetes using a free trial of the Enterprise Edition, exploring the multiple features available in this edition. For more information, see [How Does the Enterprise Edition Free Trial Work](#).

Prerequisites

Before deploying Dremio on Kubernetes, ensure you have the following:

A hosted Kubernetes environment to deploy and manage the Dremio cluster.

Each Dremio release is tested against [Amazon Elastic Kubernetes Service \(EKS\)](#), [Azure Kubernetes Service \(AKS\)](#), and [Google Kubernetes Engine \(GKE\)](#) to ensure compatibility. If you have a containerization platform built on top of Kubernetes that is not listed here, please contact your provider and the Dremio Account Team regarding compatibility.

Helm 3 installed on your local machine to run Helm commands. For installation instructions, refer to [Installing Helm](#) in the Helm documentation.

A local kubectl configured to access your Kubernetes cluster. For installation instructions, refer to [kubectl](#) in the Kubernetes documentation.

Object Storage: Amazon S3 (including S3-compatible, e.g., MinIO), Azure Storage, or Google Cloud Storage (GCS).

Step 1: Deploy Dremio

Let's start by deploying the Enterprise Edition on your hosted Kubernetes environment:

If you haven't already, [sign up for the Enterprise Edition Free Trial](#).

In the email you receive from Dremio, click a link to download the `values-overrides.yaml` file containing the deployment information and save the file locally.

Open the `values-overrides.yaml` file in an editor to make the following configurations:

For distributed storage, follow the instructions in [Configuring the Distributed Storage](#), and then return here.

For object storage, follow the instructions in [Configuring Storage for Dremio Catalog](#), and then return here.

(Optional) For the coordinator, you can adjust its default values by following the instructions in [Recommended Resources Configuration](#), and then return here.

Save the `values-overrides.yaml` file.

Open a terminal window, and start the deployment by installing Dremio's Helm chart with the following command:

Run helm install

```
helm install <your-dremio-install-release> \
  oci://quay.io/dremio/dremio-helm \
  -f <your-local-path>/values-overrides.yaml
```

Where:

`<your-dremio-install-release>` - The name that identifies your Dremio installation. For example, `dremio-1-0`.

`<your-local-path>` - The path to reach the `values-overrides.yaml` file.

Monitor the deployment using the following command:

Run kubectl to monitor pods

```
kubectl get pods
```

The deployment is complete when all pods are in the `Ready` state.

Now, access the Dremio console to interact with the platform in a user-friendly and visual way. It is a key component of the Dremio experience and is accessible through a web browser:

Run the following command in Kubernetes to find the host for the Dremio console:

Run kubectl to find the Dremio console

```
kubectl get services dremio-client
```

Depending on the value in the `TYPE` column of the output, open the Dremio console in your browser with one of the following URLs:

`https://EXTERNAL_IP:9047` - If the value in the `TYPE` column is `LoadBalancer`, use the value from the `EXTERNAL_IP` column of the output in the URL. For example, `https://8.8.8.8:9047`.

`<http://localhost:32390>` - If the value in the `TYPE` column is `NodePort`.

Follow the instructions, and enter your details.

You should have the Dremio console ready in your browser.

![Dremio console landing page](#).

To learn how to navigate the Dremio console, see [Quick Tour of the Dremio Console](#).

Step 2: Create an Engine

Engines are responsible for query execution. Each engine comprises one or more executors that perform queries and Data Manipulation Language (DML) operations by running the query execution plan and transiting data between themselves to serve queries.

To create an engine, do the following:

Click ![This is the icon that represents the Organization settings](#) in the side navigation bar to go to the Settings page.

Select **Engines** from the settings sidebar, and then click **Add Engine** on the far right.

In the New Engine dialog, enter a name for your engine. For example, `my-engine`.

Click **Add**.

You will see a new line with your engine with the **Status** as `Starting`.

Wait until the **Status** changes to `Running` for the engine to be available to serve your queries.

note

The engine you created is configured to automatically stop/start. This means that Dremio automatically stops the engine after 15 minutes of idle time to save resources. When a new query is issued, Dremio automatically starts the engine, but your query may take a bit longer to execute while the engine starts.

If you want to have the engine always running, edit the engine and uncheck the **Automatically start/stop** option.

Step 3: Add the Sample Data

Let's add the sample datasets that will be used in this Get Started guide, namely:

NYC taxi trip data - In Iceberg format, with more than 338 million records.

NYC weather data - In CSV format, with more than 4 thousand records.

Add the Datasets

Add the datasets from a sample data source, as follows:

In the Dremio console, click [!This is the icon that represents the Datasets page.](#) in the side navigation bar to go to the Datasets page.

Click [!This is the Add Source icon.](#) right next to **Sources**.

In the Add Source dialog, select `Sample Source` in the **Object Storage** section.

Format the Datasets

Now that the data source has been added, let's format the needed datasets as tables so that we can query them:

Under **Object Storage**, click the newly added `Samples` source, and then `samples.dremio.com` to see its details.

Hover over the `NYC-taxi-trips-iceberg` folder, and click [!This is the icon that represents the format folder action.](#) on the far right.

In the Folder Settings dialog, check the **Format**, verify that `Iceberg` is detected, and click **Save**.

Click [!This is the icon that represents the Datasets page.](#) in the side navigation bar, click the `Samples` source, and then `samples.dremio.com` to see its details.

Hover over the `NYC-weather.csv` file, and click [!This is the icon that represents the format file action.](#) on the far right.

In the Table Settings dialog, do the following:

For **Line Delimiter**, select `LF - Unix/Linux`.

Under **Options**, check **Extract Column Names**.

Click **Save**.

The sample data is now added, formatted, and ready to be queried.

You can validate it by clicking [!This is the icon that represents the Datasets page.](#) in the side navigation bar, then the `Samples` source, and then `samples.dremio.com` to see its details:

The icon for `NYC-taxi-trips-iceberg` is [!This is the icon that represents a formatted folder on the Datasets page.](#), which means the folder is formatted as a table.

The icon for `NYC-weather.csv` is [!This is the icon that represents the a formatted file on the Datasets page.](#), which means the file is formatted as a table.

Step 4: Create a Data Product

In this step, you will start creating a data product to explore the relationship between weather conditions and tipping behavior in taxi rides to answer the business question: "Do people tip more during taxi rides when it's raining?".

Run the Query for the Data Product

To answer the business question, you will need the average tip amount per precipitation level. For that, combine the data in the `NYC-taxi-trips-iceberg` and `NYC-weather.csv` datasets on a common field: the date.

To do this, run the SQL query that joins the two datasets, and calculates the average tip amount per precipitation level:

Click [!This is the icon that represents the SQL runner.](#) in the side navigation bar to go to the [SQL Runner](#).

Copy the SQL below, paste it in the SQL Runner, and click **Run**.

SQL to join datasets

```
SELECT AVG(tip_amount) as avg_tip_amount, prcp
FROM   Samples."samples.dremio.com"."NYC-weather.csv"
JOIN   Samples."samples.dremio.com"."NYC-taxi-trips-iceberg"
ON     (TO_CHAR(CAST(pickup_datetime AS DATE), 'YYYY-MM-DD')) = SUBSTRING(CAST("date" AS
CHAR) FROM 0 FOR 10)
GROUP BY prcp;
```

You will get the query results, as shown in the image below.

[!The result of the query to join the datasets with the average tip amount per precipitation level.](#)

Create the View for the Data Product

In Dremio, views are virtual tables based on the result set of a query. You can create

views from data that resides in any data source, folder, table, or view that you have access to. You can also share views you've created with stakeholders in your organization.

Let's create a view for the data product from the query that you ran above:

Click **Save as View** on the far right to create a view of your query that others can access.

In the Save View As dialog, enter a name for your view. For example, ``avg_tips_precipitation``.

Click **Save**.

You can see the lineage of your datasets in a graph showing all the relationships with end-to-end visibility into how data is sourced and transformed, which helps you understand the data flow and dependencies between datasets.

For your newly created view, see the lineage by selecting the **Lineage** tab at the top of the page:

!The lineage graph for the view showing how datasets are connected.

Step 5: Accelerate the Query with Reflections

In this step, you will use Reflections to accelerate queries, particularly when working with large datasets.

Enable the Reflection

Let's enable a Raw Reflection to accelerate the query of your view:

Select the **Reflections** tab at the top of the page, toggle the **Raw Reflections** switch to on, and click **Save**.

On the far right, you will see an animated spinner icon close to **Footprint**. Wait until it turns into a green checkmark, which means that your query has been accelerated.

Run the Accelerated Query

Let's now query the view and see the acceleration in action:

Click !This is the icon that represents the Datasets page., click ``avg_tips_precipitation``, and click **Run** to execute the query.

Check the execution time. It's **a sub-second query!**

!The query with a highlight in the execution time after the acceleration.

Now, go to the Jobs tab, and confirm that the query was accelerated with a Reflection.

!The details of a query showing it was accelerated with a Reflection.

You've just created a Raw Reflection and accelerated your query!

While creating a Reflection manually is a great way to understand how Dremio boosts

performance, you don't need to manage this complexity yourself in real-world environments if you use [Autonomous Reflections](#) — available exclusively in the Enterprise Edition. Dremio will automatically create, select, and maintain the most efficient Reflections for you, saving time while ensuring consistently fast performance of your queries.

And that's it! You finished the Get Started guide for the Enterprise Edition free trial.

Explore the documentation to learn more about Dremio, start using your data, build your data products, connect your client applications, and much more.

Was this page helpful?

[\[Previous](#)

[Get Started with Dremio\]](#)(/current/get-started/)[[Next](#)

[Community Edition on Docker\]](#)(/current/get-started/docker)

[Prerequisites](#)

[Step 1: Deploy Dremio](#)

[Step 2: Create an Engine](#)

[Step 3: Add the Sample Data](#)

[Add the Datasets](#)

[Format the Datasets](#)

[Step 4: Create a Data Product](#)

[Run the Query for the Data Product](#)

[Create the View for the Data Product](#)

[Step 5: Accelerate the Query with Reflections](#)

[Enable the Reflection](#)

[Run the Accelerated Query](#)

Source:
<https://docs.dremio.com/current/get-started/docker>

Version: current [26.x]

On this page

Get Started with the Community Edition on Docker

This Docker-based Get Started guide offers a simple and fast way to spin up Dremio locally with the Community Edition and explore the capabilities available in this edition.

This Docker deployment is indicated for testing and evaluation purposes and is not recommended for production usage. To try out a complete version of Dremio with enterprise-grade features, go to [Get Started with the Enterprise Edition Free Trial](#).

Prerequisites

Before you start, download and install [Docker Desktop](#).

Step 1: Deploy Dremio

Let's deploy the Dremio Community Edition on Docker:

Open your Docker Desktop.

Click >\ **Terminal** on the bottom-right of the screen, and run the following command:

Run Docker command

```
docker run \
  -p 9047:9047 -p 31010:31010 -p 45678:45678 -p 32010:32010 \
  -e DREMIO_JAVA_SERVER_EXTRA_OPTS=-Dpaths.dist=file:///opt/dremio/data/dist \
  dremio/dremio-oss
```

After a couple of minutes, the containers should be up and running, and Dremio is deployed.

Now, access the Dremio console, where you interact with the platform in a user-friendly and visual way. It is a key component of the Dremio experience and is accessible through a web browser:

In your browser, navigate to <<http://localhost:9047>>.

You will be asked to enter your details and click **Next**.

You should have the Dremio console ready in your browser.

[!Dremio console landing page](#).

To learn how to navigate the Dremio console, see [Quick Tour of the Dremio Console](#).

Step 2: Add the Sample Data

Let's add the sample datasets that will be used in this Get Started guide, namely:

NYC taxi trip data – In Iceberg format, with more than 338 million records.

NYC weather data – In CSV format, with more than 4 thousand records.

Add the Datasets

Add the datasets from a sample data source, as follows:

-

In the Dremio console, click [!This is the icon that represents the Datasets page.](#) in the side navigation bar to go to the Datasets page.

Click [!This is the Add Source icon.](#) right next to **Sources**.

In the Add Source dialog, select `Sample Source` in the **Object Storage** section.

Format the Datasets

Now that the data source has been added, let's format the needed datasets as tables so that we can query them:

Under **Object Storage**, click the newly added `Samples` source, and then `samples.dremio.com` to see its details.

Hover over the `NYC-taxi-trips-iceberg` folder, and click [!This is the icon that represents the format folder action.](#) on the far right.

In the Folder Settings dialog, check the **Format**, verify that `Iceberg` is detected, and click **Save**.

Click [!This is the icon that represents the Datasets page.](#) in the side navigation bar, click the `Samples` source, and then `samples.dremio.com` to see its details.

Hover over the `NYC-weather.csv` file, and click [!This is the icon that represents the format file action.](#) on the far right.

In the Table Settings dialog, do the following:

For **Line Delimiter**, select `LF - Unix/Linux`.

Under **Options**, check **Extract Column Names**.

Click **Save**.

The sample data is now added, formatted, and ready to be queried.

You can validate it by clicking [!This is the icon that represents the Datasets page.](#) in the side navigation bar, then the `Samples` source, and then `samples.dremio.com` to see its details:

The icon for `NYC-taxi-trips-iceberg` is [!This is the icon that represents a formatted folder on the Datasets page.](#), which means the folder is formatted as a table.

The icon for `NYC-weather.csv` is [!This is the icon that represents the a formatted file on the Datasets page.](#), which means the file is formatted as a table.

Step 3: Create a Data Product

In this step, you will start creating a data product to explore the relationship between weather conditions and tipping behavior in taxi rides to answer the business question: "Do people tip more during taxi rides when it's raining?".

Run the Query for the Data Product

To answer the business question, you will need the average tip amount per precipitation level. For that, combine the data in the `NYC-taxi-trips-iceberg` and `NYC-weather.csv` datasets on a common field: the date.

To do this, run the SQL query that joins the two datasets, and calculates the average tip amount per precipitation level:

Click [!This is the icon that represents the SQL runner.](#) in the side navigation bar to go to the [SQL Runner](#).

Copy the SQL below, paste it in the SQL Runner, and click **Run**.

SQL to join datasets

```
SELECT AVG(tip_amount) as avg_tip_amount, prcp
FROM   Samples."samples.dremio.com"."NYC-weather.csv"
JOIN   Samples."samples.dremio.com"."NYC-taxi-trips-iceberg"
ON     (TO_CHAR(CAST(pickup_datetime AS DATE), 'YYYY-MM-DD')) = SUBSTRING(CAST("date" AS
CHAR) FROM 0 FOR 10)
GROUP BY prcp;
```

You will get the query results, as shown in the image below.

[!The result of the query to join the datasets with the average tip amount per precipitation level.](#)

Create the View for the Data Product

In Dremio, views are virtual tables based on the result set of a query. You can create views from data that resides in any data source, folder, table, or view that you have access to. You can also share views you've created with stakeholders in your organization.

Let's create a view for the data product from the query that you ran above:

Click **Save as View** on the far right to create a view of your query that others can access.

On the Save View As dialog, enter a name for your view. For example, `avg_tips_precipitation`.

Click **Save**.

Your view is created and ready to be used.

Step 4: Accelerate the Query with Reflections

In this step, you will use [Reflections](#) to accelerate queries, particularly when working with large datasets.

Enable the Reflection

Let's enable a [Raw Reflection](#) to accelerate the query of your view:

Select the **Reflections** tab at the top of the page, toggle the **Raw Reflections** switch to on, and click **Save**.

On the far right, you will see an animated spinner icon close to **Footprint**. Wait until it turns into a green checkmark, which means that your query has been accelerated.

Run the Accelerated Query

Let's now query the view and see the acceleration in action:

Click !This is the icon that represents the Datasets page., click ``avg_tips_precipitation``, and click **Run** to execute the query.

Check the execution time. It's **a sub-second query!**

!The query with a highlight in the execution time after the acceleration.

Now, go to the Jobs tab, and confirm that the query was accelerated with a Reflection.

!The details of a query showing it was accelerated with a Reflection.

You've just created a Raw Reflection and accelerated your query!

While creating a Reflection manually is a great way to understand how Dremio boosts performance, you don't need to manage this complexity yourself in real-world environments if you use Autonomous Reflections — available exclusively in the Enterprise Edition. Dremio will automatically create, select, and maintain the most efficient Reflections for you, saving time while ensuring consistently fast performance of your queries.

And that's it! You finished the Get Started guide for the Community Edition on Docker.

For a more complete and full-featured experience with Dremio, sign up for the Enterprise Edition free trial on the Dremio website, and follow the steps in Get Started with the Enterprise Edition Free Trial.

Was this page helpful?

[Previous

Enterprise Edition Free Trial](/current/get-started/kubernetes-trial)[Next

Quick Tour of the Console](/current/get-started/quick_tour)

Prerequisites

Step 1: Deploy Dremio

Step 2: Add the Sample Data

Add the Datasets

Format the Datasets

Step 3: Create a Data Product

Run the Query for the Data Product

Create the View for the Data Product

Step 4: Accelerate the Query with Reflections

Enable the Reflection

Run the Accelerated Query

Source:
https://docs.dremio.com/current/get-started/quick_tour

Version: current [26.x]

On this page

Quick Tour of the Console

This section walks you through the Dremio console and how to best use the different capabilities to get to insights quickly and how to manage your data products.

The Dremio console has two main pages:

Datasets page: Provides a view of your data products and underlying source tables. You can discover and explore your data on this page. For a quick tour of the Datasets page, see [Datasets Page Quick Tour](#).

SQL Runner: An easy to use editor to query data and create data products. For a quick tour of the SQL Runner, see [SQL Runner Quick Tour](#).

Datasets Page

When you work on the Datasets page, there are different components that you can use to manage your data. The largest component is the **Data** panel, which is used to explore the spaces and sources in your data catalog, as shown in this image:

!This screenshot is displaying the Data panel.

Location	Description
1	By default, you have a home space that you can further organize by creating a hierarchy of folders, and you can create additional spaces.
2	A space is a directory in which views are saved. Spaces provide a way to group datasets by common themes such as a project, purpose, department, or geographic region.
3	A source is a data lake or external source (such as a relational database) that you can connect to Dremio.
4	The title indicates that the Samples data lake is open and lists the contents of the sample source. A source also consists of layers, so if you expand a data source, you will find datasets and data types within the datasets.
5	A dataset is a collection of data. The datasets stored in files can be in many different formats, and to run SQL queries against data in different formats, you can create tables and views. By default, when you click on a dataset, the SQL editor is

opened on the SQL Runner page with a ``SELECT * FROM <dataset_name>`` statement. To get a preview of the query, click **Run** or **Preview**. If you would rather click directly on a dataset to see or edit the definition, see [Preferences](#) for modifying this setting. |

Opening Datasets


If you have privileges to modify the table or view, you will have the option of viewing and editing the table or view definition. When viewing or editing a table or view, you are directed to the **Data** tab by default, which shows the definitions of the table or view. For more options, check out the other tabs:

!This is a screenshot showing the Datasets page.

- | Location | Description |
- | --- | --- |
- | 1 | [Details](#) shows the columns in a dataset and lets you add information about a specific dataset in its wiki. You can add searchable labels, which enhances team collaboration because other users can search the labels to trace a specific dataset. |
- | 2 | [Lineage](#) is a graph of the dataset, showing its data source, parent datasets, and children datasets. |
- | 3 | [Reflections](#) are physically optimized representations of source data. |

Tabs are made visible based on the privileges that you have.

SQL Runner

The SQL Runner is where you run queries on your datasets and get results. To navigate to the SQL Runner, click  in the side navigation bar. The main components of the SQL Runner are highlighted below:

!The main components of the SQL Runner are highlighted.

caution

Dremio's query engine intentionally ignores any file or folder if the filename or folder name starts with a period (".") or an underscore ("_").

1. Data

The **Data** panel is used to explore your data catalog, which includes [sources](#), [tables](#), and [views](#). For catalog objects that you use frequently, you can star the objects to make them easier to access from the panel.

To add a dataset into the SQL editor, go to the data source. Use the left caret > to expand the source view. Locate the dataset that you would like to use within the query. Click the + button or drag and drop the data into the SQL editor.

2. Scripts

You can save your SQL as a script if you have drafts or SQL statements that you run frequently. Each saved script has a name, when the script was created or modified, and the context that was set for the editor.

In the **Scripts** panel, you can:

Open a script in the SQL editor

Rename a script

Delete a script

Share a script by [granting privileges](#)

Search your set of scripts by name

Sort scripts by name or date

3. Run Mode

Running the query routes it to the engine and returns the complete result set. Dremio's query engine intentionally ignores any file or folder if the filename or folder name starts with a period (".") or an underscore ("_").

caution

If the engine scaled down, the startup time will take about two minutes.

note

Sometimes ``COUNT(*)`` and ``SELECT`` query results do not match because the result of queries run in the Dremio app has a threshold of one million. Depending on the number of threads (minor fragments) and how data is distributed among those threads, Dremio could truncate results before reaching the threshold. Each individual thread stops processing after returning a number of records equal to ``threshold/# of threads``. For example, a query runs with 10 threads and should return 800,000 records, but one of the threads is responsible for 400,000 records. The per-thread threshold is 100,000, so that thread will only return 100,000 records and you will only see 500,000 records in the output. When results are truncated, the Dremio app will display a warning that the results are not complete and users can execute the query using JDBC/ODBC to get complete results.

note

Known issue: Running a ``USE`` statement will not update the context that is set in the SQL Runner.

4. Preview Mode

Executing a preview returns a subset of rows in the result set. Like the run mode, running the preview job will route the query to the selected engine, although the preview mode runs a subset of your results in less time.

5. SQL Editor

The SQL editor is where you create and edit queries to get insight from your data. In the top-right corner of the SQL editor, you'll find:

- a. Create a new tab by clicking **+** next to the other tabs. Because a tab is defined by a session, a new tab is automatically saved as a script and named as the date and time that the session was created, such as `Nov 3, 2023, 10:19:57 AM`.
- b. Grant script privileges to share a saved script with others in your organization.
- c. Save your SQL as a script or as a view. You can save a script even if there are syntax errors. Saving a new view requires valid syntax, and there can be only one SQL statement in the editor. When you save the script as a view, the tab will remain open in the SQL Runner and the Edit dataset page for the view will open in a new browser tab.
- d. Set a **Context** for a session to run queries without having to qualify the referenced objects.
- e. Use **fx** to see a list of functions supported by Dremio along with a short description and syntax of each function. Click on the **+** button or drag and drop the function template into the SQL editor. For more information on supported SQL, see the SQL Reference.
- f. Toggle the **dark/light mode** to change the theme of the SQL editor.
- g. Enable autocomplete to receive suggestions for SQL keywords, catalog objects, and functions while you are constructing SQL statements. Suggestions depend on the context set in the SQL Runner. To enable or disable the autocomplete feature, see Dremio Preferences.
- h. Click the keyboard button to see the shortcuts for the SQL Runner. For a list of shortcuts, see Keyboard Shortcuts.

Syntax Error Highlighting

Before you run a query, make sure to fix any syntax errors that have been detected in your query.

The SQL editor automatically checks for syntax errors, and every detected error is marked with a red wavy underline. If you hover over the error, you'll see a message stating whether the error is the result of a token that is missing, unexpected, unrecognized, or extraneous in the query.

Running Multiple Queries

You can run multiple queries in the SQL editor by using a semicolon to separate each statement. To run all of the queries in the SQL editor, simply click **Run**. The results of each query will be shown in the same order that the queries are constructed:

When you have multiple queries, you can also select a subset to run. If you select one or more queries and then click **Run**, only the selected queries will run accordingly, as shown below:

6. Result Set Actions

Above the top-right corner of the result set, you will find these actions:

- a. Download the result set as a JSON, CSV, or Parquet file. By default, downloading writes out the results from the last run of the query into the download file. To trigger a rerun of queries for downloads, see the [Rerun on download preference](#).
- b. Copy result set to a clipboard. If the result set is too large, then download the table content to get the complete results.

note

The option to download as a CSV file is not available if the result set includes one or more columns that have complex datatypes (ie., a union, map, or array). Column headers for the results table indicate complex types with this icon:

 command, which is why compute resources are required.

Page 496

and copying results from the project.

7. Execution State

The execution state will show you the type of job that was run, the number of records, and the amount of time that it took to run the query. When you click on the linked job, a Job Overview page opens in a new tab, providing a summary, total execution time, Reflections data, job results, and more details. If a job took too long or failed, [viewing the job overview](#) can help you troubleshoot what actually happened.

8. Transformations

Transformations can be applied to data. Using the following no-code UI flows automatically updates the SQL in the SQL editor:

Add Field

Group By

Join

Filter Columns

If you are using the preview mode, transformations use a subset of the results and may not provide a complete profile of the result set. It may show null or incomplete values in the dataset as a result of joining, grouping, or calculating fields. You may encounter an error showing "no rows returned due to the LIMIT logic" or "more rows returned due to an outer join".

9. Results Table

The results of the query are shown in a table format. You can edit a table column by clicking directly on the column title to open a dropdown of edit options, which include sorting results, converting data types, renaming columns, and calculating fields.

You can edit a result value directly if you click and drag your cursor over the result value, which opens a dropdown of available edit options such as to replace, keep only, exclude, or copy the result value.

tip

Downloading large result sets could produce delays and errors. If you encounter these issues, create smaller views that summarize the results. You can then run queries on these smaller views and download the results.

Additional Resources

Find out more about Dremio by enrolling in the [Dremio Fundamentals course in Dremio University](#).

Was this page helpful?

[Previous

Community Edition on Docker](/current/get-started/docker)[Next

What is Dremio?](/current/what-is-dremio/)

Datasets Page

Opening Datasets

SQL Runner

Data

Scripts

Run Mode

Preview Mode

SQL Editor

Result Set Actions

Execution State

Transformations

Results Table

Additional Resources

Source:
https://docs.dremio.com/current/get-started/quick_tour/

Version: current [26.x]

On this page

Quick Tour of the Console

This section walks you through the Dremio console and how to best use the different capabilities to get to insights quickly and how to manage your data products.

The Dremio console has two main pages:

Datasets page: Provides a view of your data products and underlying source tables. You can discover and explore your data on this page. For a quick tour of the Datasets page, see [Datasets Page Quick Tour](#).

SQL Runner: An easy to use editor to query data and create data products. For a quick tour of the SQL Runner, see [SQL Runner Quick Tour](#).

Datasets Page

When you work on the Datasets page, there are different components that you can use to manage your data. The largest component is the **Data** panel, which is used to explore the spaces and sources in your data catalog, as shown in this image:

!This screenshot is displaying the Data panel.

- | Location | Description |
- | --- | --- |
- | 1 | By default, you have a home space that you can further organize by creating a hierarchy of folders, and you can create additional spaces. |
- | 2 | A space is a directory in which views are saved. Spaces provide a way to group datasets by common themes such as a project, purpose, department, or geographic region. |
- | 3 | A source is a data lake or external source (such as a relational database) that you can connect to Dremio. |
- | 4 | The title indicates that the Samples data lake is open and lists the contents of the sample source. A source also consists of layers, so if you expand a data source, you will find datasets and data types within the datasets. |
- | 5 | A dataset is a collection of data. The datasets stored in files can be in many different formats, and to run SQL queries against data in different formats, you can create tables and views. By default, when you click on a dataset, the SQL editor is opened on the SQL Runner page with a ``SELECT * FROM <dataset_name>`` statement. To get a preview of the query, click **Run** or **Preview**. If you would rather click directly on a dataset to see or edit the definition, see Preferences for modifying this setting. |

Opening Datasets


If you have privileges to modify the table or view, you will have the option of viewing and editing the table or view definition. When viewing or editing a table or view, you are directed to the **Data** tab by default, which shows the definitions of the table or view. For more options, check out the other tabs:

!This is a screenshot showing the Datasets page.

- | Location | Description |
- | --- | --- |
- | 1 | Details shows the columns in a dataset and lets you add information about a specific dataset in its wiki. You can add searchable labels, which enhances team collaboration because other users can search the labels to trace a specific dataset. |
- | 2 | Lineage is a graph of the dataset, showing its data source, parent datasets, and children datasets. |
- | 3 | Reflections are physically optimized representations of source data. |

Tabs are made visible based on the privileges that you have.

SQL Runner

The SQL Runner is where you run queries on your datasets and get results. To navigate to the SQL Runner, click  in the side navigation bar. The main components of the SQL Runner are highlighted below:

!The main components of the SQL Runner are highlighted.

caution

Dremio's query engine intentionally ignores any file or folder if the filename or folder name starts with a period (".") or an underscore ("_").

1. Data

The **Data** panel is used to explore your data catalog, which includes sources, tables, and views. For catalog objects that you use frequently, you can star the objects to make them easier to access from the panel.

To add a dataset into the SQL editor, go to the data source. Use the left caret > to expand the source view. Locate the dataset that you would like to use within the query. Click the + button or drag and drop the data into the SQL editor.

2. Scripts

You can save your SQL as a script if you have drafts or SQL statements that you run frequently. Each saved script has a name, when the script was created or modified, and the context that was set for the editor.

In the **Scripts** panel, you can:

Open a script in the SQL editor

Rename a script

Delete a script

Share a script by granting privileges

Search your set of scripts by name

Sort scripts by name or date

3. Run Mode

Running the query routes it to the engine and returns the complete result set. Dremio's query engine intentionally ignores any file or folder if the filename or folder name starts with a period (".") or an underscore ("_").

caution

If the engine scaled down, the startup time will take about two minutes.

note

Sometimes ``COUNT(*)`` and ``SELECT`` query results do not match because the result of queries run in the Dremio app has a threshold of one million. Depending on the number of threads (minor fragments) and how data is distributed among those threads, Dremio could truncate results before reaching the threshold. Each individual thread stops processing after returning a number of records equal to ``threshold/# of threads``. For

example, a query runs with 10 threads and should return 800,000 records, but one of the threads is responsible for 400,000 records. The per-thread threshold is 100,000, so that thread will only return 100,000 records and you will only see 500,000 records in the output. When results are truncated, the Dremio app will display a warning that the results are not complete and users can execute the query using JDBC/ODBC to get complete results.

note

Known issue: Running a ``USE`` statement will not update the context that is set in the SQL Runner.

4. Preview Mode

Executing a preview returns a subset of rows in the result set. Like the run mode, running the preview job will route the query to the selected engine, although the preview mode runs a subset of your results in less time.

5. SQL Editor

The SQL editor is where you create and edit queries to get insight from your data. In the top-right corner of the SQL editor, you'll find:

- a. Create a new tab by clicking **+** next to the other tabs. Because a tab is defined by a session, a new tab is automatically saved as a script and named as the date and time that the session was created, such as ``Nov 3, 2023, 10:19:57 AM``.
- b. Grant script privileges to share a saved script with others in your organization.
- c. Save your SQL as a script or as a view. You can save a script even if there are syntax errors. Saving a new view requires valid syntax, and there can be only one SQL statement in the editor. When you save the script as a view, the tab will remain open in the SQL Runner and the Edit dataset page for the view will open in a new browser tab.
- d. Set a **Context** for a session to run queries without having to qualify the referenced objects.
- e. Use **fx** to see a list of functions supported by Dremio along with a short description and syntax of each function. Click on the **+** button or drag and drop the function template into the SQL editor. For more information on supported SQL, see the [SQL Reference](#).
- f. Toggle the **dark/light mode** to change the theme of the SQL editor.
- g. Enable autocomplete to receive suggestions for SQL keywords, catalog objects, and functions while you are constructing SQL statements. Suggestions depend on the context set in the SQL Runner. To enable or disable the autocomplete feature, see [Dremio Preferences](#).
- h. Click the keyboard button to see the shortcuts for the SQL Runner. For a list of shortcuts, see [Keyboard Shortcuts](#).

Syntax Error Highlighting

Before you run a query, make sure to fix any syntax errors that have been detected in your query.

The SQL editor automatically checks for syntax errors, and every detected error is marked with a red wavy underline. If you hover over the error, you'll see a message stating whether the error is the result of a token that is missing, unexpected, unrecognized, or extraneous in the query.

Running Multiple Queries

You can run multiple queries in the SQL editor by using a semicolon to separate each statement. To run all of the queries in the SQL editor, simply click **Run**. The results of each query will be shown in the same order that the queries are constructed:

When you have multiple queries, you can also select a subset to run. If you select one or more queries and then click **Run**, only the selected queries will run accordingly, as shown below:

6. Result Set Actions

Above the top-right corner of the result set, you will find these actions:

- Download the result set as a JSON, CSV, or Parquet file. By default, downloading writes out the results from the last run of the query into the download file. To trigger a rerun of queries for downloads, see the [Rerun on download preference](#).
- Copy result set to a clipboard. If the result set is too large, then download the table content to get the complete results.

note

The option to download as a CSV file is not available if the result set includes one or more columns that have complex datatypes (ie., a union, map, or array). Column headers for the results table indicate complex types with this icon:

The download process runs a CREATE TABLE AS SELECT (CTAS) command, which is why compute resources are required.

The download and copy results features can be enabled or disabled for a project in [Dremio Preferences](#). Disabling this in a project will prevent all users from downloading and copying results from the project.

7. Execution State

The execution state will show you the type of job that was run, the number of records, and the amount of time that it took to run the query. When you click on the linked job, a Job Overview page opens in a new tab, providing a summary, total execution time, Reflections data, job results, and more details. If a job took too long or failed, [viewing the job overview](#) can help you troubleshoot what actually happened.

8. Transformations

Transformations can be applied to data. Using the following no-code UI flows automatically updates the SQL in the SQL editor:

Add Field

Group By

Join

Filter Columns

If you are using the preview mode, transformations use a subset of the results and may not provide a complete profile of the result set. It may show null or incomplete values in the dataset as a result of joining, grouping, or calculating fields. You may encounter an error showing "no rows returned due to the LIMIT logic" or "more rows returned due to

an outer join".

9. Results Table

The results of the query are shown in a table format. You can edit a table column by clicking directly on the column title to open a dropdown of edit options, which include sorting results, converting data types, renaming columns, and calculating fields.

You can edit a result value directly if you click and drag your cursor over the result value, which opens a dropdown of available edit options such as to replace, keep only, exclude, or copy the result value.

tip

Downloading large result sets could produce delays and errors. If you encounter these issues, create smaller views that summarize the results. You can then run queries on these smaller views and download the results.

Additional Resources

Find out more about Dremio by enrolling in the [Dremio Fundamentals course in Dremio University](#).

Was this page helpful?

[\[Previous](#)

[Community Edition on Docker\]\(/current/get-started/docker\)\[Next](#)

[What is Dremio?\]\(/current/what-is-dremio/\)](#)

[Datasets Page](#)

[Opening Datasets](#)

[SQL Runner](#)

[Data](#)

[Scripts](#)

[Run Mode](#)

[Preview Mode](#)

[SQL Editor](#)

[Result Set Actions](#)

[Execution State](#)

[Transformations](#)

[Results Table](#)

Source: dremio-software-monitoring.md

Dremio Software - Monitoring

Source:
<https://docs.dremio.com/current/admin/monitoring/>

Version: current [26.x]

On this page

Monitoring

As an administrator, you can monitor logs, usage, system telemetry, [jobs](#), and [Dremio nodes](#).

As the [Dremio Shared Responsibility Models](#) outline, monitoring is a shared responsibility between Dremio and you. The Shared Responsibility Models lay out Dremio's responsibilities for providing monitoring technologies and logs and your responsibilities for implementation and use.

Logs

Logs are primarily for troubleshooting issues and monitoring the health of the deployment.

note

By default, Dremio uses the following locations to write logs:

Tarball - ``<DREMIO_HOME>/log``

RPM - ``/var/log/dremio``

Kubernetes - ``/opt/dremio/log``

Log Types

Log Type	Description
----------	-------------

---	---
-----	-----

Audit	The <code>audit.json</code> file tracks all activities that users perform within Dremio. For details, see Audit Logging .
-------	---

System	The following system logs are enabled by default: * <code>access.log</code> : HTTP access log for the Dremio web server; generated by coordinator nodes only. * <code>server.gc</code> : Garbage collection log. * <code>server.log</code> and <code>json/server.json</code> : Server logs generated in a text format (<code>server.log</code>) and json format (<code>json/server.json</code>). Users granted the <code>ADMIN</code> role can disable one of these formats. * <code>server.out</code> : Log for Dremio
--------	---

daemon standard out. * ``metadata_refresh.log``: Log for refreshing metadata. * ``tracker.json``: Tracker log. * ``vacuum.json``: Log for the files scanned and deleted by ``VACUUM CATALOG`` and ``VACUUM TABLE`` commands. |

| Query | Query logging is enabled by default. The ``queries.json`` file contains the log of completed queries; it does not include queries currently in planning or execution. You can retrieve the same information that is in ``queries.json`` using the ``sys.jobs_recent`` system table. Query logs include the following information: * ``queryId``: Unique ID of the executed query. * ``queryText``: SQL query text. * ``start``: Start time of the query. * ``finish``: End time of the query. * ``outcome``: Whether the query was completed or failed. * ``username``: User who executed the query. * ``commandDescription``: Type of the command; may be a regular SQL query execution job or another SQL command. The query log may contain additional information depending on your Dremio configuration. |

| Warning | The ``hive.deprecated.function.warning.log`` file contains warnings for Hive functions that have been deprecated. To resolve warnings that are listed in this file, replace deprecated functions with a [supported function](#). For example, to resolve a warning that mentions ``NVL``, replace ``NVL`` with ``COALESCE``. |

Retrieving Logs from the Dremio Console Enterprise

Retrieve logs for Kubernetes deployments in the Dremio console at **Settings > Support > Download Logs**.

Prerequisites

You must be using Dremio 25.1+. Log collection is powered by Dremio Diagnostics Collector (DDC).

You must have the EXPORT DIAGNOSTICS privilege to view **Download Logs** options in **Settings > Support**.

Downloading Logs

To download logs:

In the Dremio console, navigate to **Settings > Support > Download Logs** and click **Start collecting data**.

note

You may store a maximum of three log bundles. Delete log bundles as needed to start a new log collection if you reach the maximum.

We recommend the default ``Light`` collection, which provides 7 days of logs and completed queries in the ``queries.json`` file, for troubleshooting most issues. For more complex issues, select the ``Standard`` collection, which provides 7 days of logs and 28 days of completed queries in the ``queries.json`` file.

When Dremio completes log collection, the log bundle appears in a list below **Start collecting data**. To download a log bundle, click **Download** next to the applicable bundle. Log bundles are available to download for 24 hours.

Logging in Kubernetes

By default, all logs are written to a persisted volume mounted at ``/opt/dremio/log``.

To disable logging, set ``writeLogsToFile: false`` in the ``values-overrides.yaml`` configuration file either globally or individually for each ``coordinator`` and ``executor`` parent. For more information, see [Configuring Your Values](#).

Using the Container Console

All logs are written to the container's console (stdout) simultaneously. These logs can be monitored using a ``kubectl`` command:

Command for viewing logs using `kubectl logs`

```
kubectl logs [-f] [container-name]
```

Use the ``-f`` flag to continuously print new log entries to your terminal as they are generated.

You can also write logs to a file on disk in addition to stdout. Read [Writing Logs to a File](#) for details.

Using the AKS Container

Azure provides integration with AKS clusters and Azure Log Analytics to monitor container logs. This is a standard practice that puts infrastructure in place to aggregate logs from containers into a central log store to analyze them.

AKS log monitoring is useful for the following reasons:

Monitoring logs across lots of pods can be overwhelming.

When a pod (for example, a Dremio executor) crashes and restarts, only the logs from the last pod are available.

If a pod is crashing regularly, the logs are lost, which makes it difficult to analyze the reasons for the crash.

For more information regarding AKS, see [Azure Monitor features for Kubernetes monitoring](#).

Enabling Log Monitoring

You can enable log monitoring when creating an AKS cluster or after the cluster has been created.

Once logging is enabled, all your container ``stdout`` and ``stderr`` logs are collected by the infrastructure for you to analyze.

While creating an AKS cluster, enable container monitoring. You can use an existing Log Analytics workspace or create a new one.

In an existing AKS cluster where monitoring was not enabled during creation, go to **Logs on the AKS cluster** and enable it.

Viewing Container Logs

To view all the container logs:

Go to **Monitoring > Logs**.

Use the filter option to see the logs from the containers that you are interested in.

Usage

Monitoring usage across your cluster makes it easier to observe patterns, analyze the resources being consumed by your data platform, and understand the impact on your users.

Catalog Usage Enterprise

Go to **Settings > Monitor** to view your catalog usage. You must be a member of the `ADMIN` role to access the Monitor page. When you open the Monitor page, you are directed to the Catalog Usage tab by default where you can see the following metrics:

Top 10 most queried datasets and how often the jobs on the dataset were accelerated

Top 10 most queried spaces and source folders

note

A source can be listed in the top 10 most queried spaces and source folders if the source contains a child dataset that was used in the query (for example, `postgres.accounts`). Queries of datasets in sub-folders (for example, `s3.mybucket.iceberg_table`) are classified by the sub-folder and not the source.

All datasets are assessed in the metrics on the Monitor page except for datasets in the system tables, the information schema, and home spaces.

The metrics on the Monitor page analyze only user queries. Refreshes of data Reflections and metadata refreshes are excluded.

Jobs Enterprise

Go to **Settings > Monitor > Jobs** to open the Jobs tab. You must be a member of the `ADMIN` role to access the Monitor page. The Jobs tab shows an aggregate view of the following metrics for the jobs that are running on your cluster:

Total job count over the last 24 hours and the relative rate of failure/cancelation

Top 10 most active users based on the number of jobs they ran

Total jobs accelerated, total job time saved, and average job speedup from Autonomous Reflections over the past month.

Total number of jobs accelerated by autonomous and manual Reflections over time

Total number of completed and failed jobs over time

Jobs (completed and failed) grouped by the queue they ran on

Percentage of time that jobs spent in each state

Top 10 longest running jobs

To view all jobs and the details of specific jobs, see [Viewing Jobs](#).

Resources Enterprise

Go to **Settings** > **Monitor** > **Resources** to open the Resources tab. You must be a member of the `ADMIN` role to access the Monitor page. The Resources tab shows an aggregate view of the following metrics for the jobs and nodes running on your cluster:


Percentage of CPU and memory utilization for each coordinator and executor node

Top 10 most CPU and memory intensive jobs

Number of running executors

Cluster Usage

Dremio displays the number of unique users who executed jobs on that day and the number of executed jobs.

Hover over  [Icon represents help](#) in the side navigation bar.

Click **About Dremio** in the menu.

Click the **Cluster Usage Data** tab.

 "Viewing Cluster Usage")

System Telemetry

Dremio exposes system telemetry metrics in Prometheus format by default. It is not necessary to configure an exporter to collect the metrics. Instead, you can specify the host and port number where metrics are exposed in the [dremio.conf](#) file and scrape the metrics with any Prometheus-compliant tool.

To specify the host and port number where metrics are exposed, add these two properties to the `dremio.conf` file:

`services.web-admin.host`: set to the desired host address (typically `0.0.0.0` or the IP address of the host where Dremio is running).

`services.web-admin.port`: set to any desired value that is greater than `1024`.

For example:

Example host and port settings in dremio.conf

```
services.web-admin.host: "127.0.0.1"
services.web-admin.port: 9090
```

Restart Dremio after you update the `dremio.conf` file to make sure your changes take effect.

Access the exported Dremio system telemetry metrics at `http://<yourHost>:<yourPort>/metrics`.

For more information about Prometheus metrics, read [Types of Metrics](#) in the Prometheus documentation.

Was this page helpful?

[Previous

Workload Management](/current/admin/workloads/workload-management)[Next

Viewing Jobs](/current/admin/monitoring/jobs/)

Logs

Log Types

Retrieving Logs from the Dremio Console Enterprise

Logging in Kubernetes

Usage

Catalog Usage Enterprise

Jobs Enterprise

Resources Enterprise

Cluster Usage

System Telemetry

Source:
<https://docs.dremio.com/current/admin/monitoring/jobs/>

Version: current [26.x]

On this page

Viewing Jobs

All jobs run in Dremio are listed on a separate page, showing the job ID, type, status, and other attributes.

To navigate to the Jobs page, click [!This is the icon that represents the Jobs page.](#) in the side navigation bar.

Search Filters and Columns

By default, the Jobs page lists the jobs run within the last 30 days and the jobs are filtered by **UI, External Tools** job types. To change these defaults for your account, you can filter on values and manage columns directly on the Jobs page, as shown in this image:

!This is a screenshot showing the main components of the Jobs page.

- a. **Search Jobs** by typing the username or job ID.
- b. **Start Time** allows you to pick the date and time at which the job began.
- c. **Status** represents one or more job states. For descriptions, see Job States and Statuses.
- d. **Attribute** includes Accelerator, AI agent, AI function, Downloads, External Tools, Internal, MCP, and UI. For descriptions, see Query Types in the Job Attributes.
- e. **User** can be searched by typing the username or checking the box next to the username in the dropdown.
- f. **Manage Columns** by checking the boxes next to additional columns that you want to see in the Jobs list. The grayed out checkboxes show the columns that are required by default. You can also rearrange the column order by clicking directly on a column to drag and drop.

Job Attributes

Each job has the following attributes, which can appear as columns in the list of jobs:

Attribute	Description
Accelerated	A lightning bolt icon in a row indicates that the job ran a query that was accelerated.
Dataset	The dataset queried, if one was queried.
Duration	The length of time (in seconds) that a job required from start to completion.
Job ID	A universally unique identifier.
Planner Cost Estimate	A cost estimate calculated by Dremio based on an evaluation of the resources that to be used in the execution of a query. The number is not in units, and is intended to give a an idea of the cost of executing a query relative to the costs of executing other queries. Values are derived by adding weighted estimates of required I/O, memory, and CPU load. In reported values, K = thousand, M = million, B = billion, and T = trillion. For example, a value of 12,543,765,321 is reported as 12.5B.
Planning Time	The length of time (in seconds) in which the query optimizer planned the execution of the query.
Query Type	Represents one of the following query types: * UI - queries issued from the SQL Runner in the Dremio UI. * External Tools - queries from client applications, such as Microsoft Power BI, Superset, Tableau, other third-party client applications, and custom applications. * Accelerator - queries related to creating, maintaining, and removing Reflections. * Internal - queries that Dremio submits for internal operations. * Downloads - queries used to download datasets. * MCP - queries issued from the Dremio MCP Server. * AI agent - queries issued from Dremio's AI Agent. * AI function

- queries that call AI functions. |
- | Queue | Dremio provides the following generic queues as a starting point for customization: * High Cost Reflections * High Cost User Queries * Low Cost Reflections * Low Cost User Queries * UI Previews |
- | Rows Returned | Number of output records. |
- | Rows Scanned | Number of input records. |
- | SQL | The SQL query that was submitted for the job. |
- | Start Time | The date and time which the job began. |
- | Status | An icon that represents one or more job states. This column is automatically shown at the start of each row. For descriptions, see [Job states and statuses](#). |
- | User | Username of the user who ran the query and initiated the job. |

Job States and Statuses

Each job passes through a sequence of states until it is complete, though the sequence can be interrupted if a query is canceled or if there is an error during a state. In this diagram, the states that a job passes through are in white, and the possible end states are in dark gray.

This table lists the statuses that the UI lets you filter on and shows how they map to the states:

Icon	Status	State	Description
---	---	---	---
	Setup	Pending	Represents a state where the query is waiting to be scheduled on the query pool.
	Metadata Retrieval		Represents a state where metadata schema is retrieved and the SQL command is parsed.
	Planning		Represents a state where the following are done: * Physical and logical planning * Reflection matching * Partition metadata retrieval * Mapping the query to a queue based workload management rules * Pick the engine associated with the query to run the query.
	Engine Start	Engine Start	Represents a state where the engine starts if it has stopped. If the engine is stopped, it takes time to restart for the executors to be active. If the engine is already started, then this state does not have a duration.
	Queued	Queued	Represents a state where a job is queued. Each queue has a limit of concurrent queries. If the queries in progress exceed the concurrency limit, the query should wait in the queue until the jobs in progress complete.
	Running	Execution Planning	Represents a state where executor nodes are selected from the chosen engine to run the query, and work is distributed to each executor.
	Running		Represents a state where executor nodes execute and complete the fragments assigned to them. Typically, most queries spend more time in this state.
	Starting		Represents a state where the query is starting up.
	Canceled	Canceled	Represents a terminal state that indicates that the query is canceled by the user or an intervention in the system.
	Completed	Completed	Represents a terminal state that indicates that the query is successfully completed.
	Failed	Failed	Represents a terminal state that indicates that the query has failed due to an error.

Was this page helpful?

Source:
<https://docs.dremio.com/current/admin/monitoring/dremio-nodes>

Version: current [26.x]

On this page

Monitoring Dremio Nodes

There are various approaches for operational monitoring of Dremio nodes. This topic discusses both:

Prometheus metrics, which can be leveraged with tools like Grafana to ensure the stability and performance of Dremio deployments.

``queries.json``, a log file generated by Dremio, which can be used to calculate various service-level agreements (SLAs) related to query performance.

While these two datasets can be used in similar ways, Prometheus metrics are less granular than ``queries.json``—the latter allows you to drill down into which specific kinds of queries or users are experiencing SLA breaches.

Enabling Node Metrics

Dremio enables node monitoring by default. Starting in Dremio 26.0, each node in the cluster exposes Prometheus metrics via the ``/metrics`` endpoint on port 9010.

Available Prometheus Metrics

The following table describes the Prometheus metrics provided by Dremio and specifies which Dremio node roles support them:

Metric Name	Description	Main Coordinator	Scale-out Coordinator	Executor
---	---	---	---	---
<code>`jobs_active`</code>	Gauge showing the number of currently active jobs	Yes	Yes	No
<code>`jobs_total`</code>	Counter of total jobs submitted, categorized by the type of query	Yes	Yes	No
<code>`jobs.failed`</code>	Counter of failed jobs categorized by query types	Yes	No	No
<code>`jobs.waiting`</code>	Gauge of currently waiting jobs categorized by queue	Yes	No	No
<code>`dremio.memory.jvm_direct_current`</code>	Total direct memory (in bytes) given to the JVM			

| Yes | Yes | Yes |

| `memory.heap.committed` | Committed heap memory as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.heap.init` | Initialized heap memory as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.heap.max` | Maximum amount of heap memory that can be allocated as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.heap.usage` | Ratio of used heap memory to max heap memory, as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.heap.used` | Amount of used heap memory, as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.non-heap.committed` | Amount of non-heap memory committed, as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.non-heap.init` | Initialized non-heap memory as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.non-heap.max` | Maximum amount of non-heap memory that can be allocated, as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.non-heap.usage` | Ratio of used non-heap memory to max non-heap memory, as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.non-heap.used` | Amount of used non-heap memory, as described in [Class MemoryUsage](#) in the Oracle documentation | Yes | Yes | Yes |

| `memory.total.committed` | Sum of heap and non-heap committed memory (in bytes) | Yes | Yes | Yes |

| `memory.total.init` | Sum of heap and non-heap initialized memory (in bytes) | Yes | Yes | Yes |

| `memory.total.max` | Sum of the heap and non-heap max memory (in bytes) | Yes | Yes | Yes |

| `memory.total.used` | Sum of the heap and non-heap used memory (in bytes) | Yes | Yes | Yes |

| `reflections.failed` | Scheduled Reflections that have failed and won't be retried | Yes | No | No |

| `reflections.unknown` | Reflections for which an error occurred in the Reflection manager and that could not be retried | Yes | No | No |

| `reflections.active` | Currently active Reflections | Yes | No | No |

| `reflections.refreshing` | Reflections that are currently refreshing or pending a refresh | Yes | No | No |

| `reflections.manager_sync` | Time taken to run Reflection management | Yes | Yes | No |

| `threads.blocked.count` | Gauge of currently blocked threads | Yes | Yes | Yes |

| `threads.count` | Gauge of active and idle threads | Yes | Yes | Yes |

| `threads.daemon.count` | Number of currently available active daemon threads | Yes | Yes | Yes |

| `threads.deadlock.count` | Number of currently deadlocked threads | Yes | Yes | Yes |

| `threads.new.count` | Current number of threads in new state (not yet started) | Yes | Yes | Yes |

| `threads.runnable.count` | Current number of threads in runnable state (executing) | Yes | Yes | Yes |

| `threads.terminated.count` | Current number of threads in the terminated state (completed execution) | Yes | Yes | Yes |

| `threads.timed_waiting.count` | Current number of threads in the timed_waiting state | Yes | Yes | Yes |

| `threads.waiting.count` | Current number of threads in the waiting state | Yes | Yes | Yes |

Yes |

| `jvm.gc.overhead.percent` | An approximate percentage of CPU time used by garbage collection activities | Yes | Yes | No |

Parameters to Monitor for Scaling Capacity

The following parameters, derived from the `queries.json`, can help identify when additional engines or vertical scaling are needed to maintain performance.

Query Execution Errors

By reviewing the `outcomeReason` field in `queries.json`, you can identify resource-related issues and take proactive steps, such as scaling engines or redistributing workloads, to maintain performance and stability.

Error Type (`outcomeReason`)	Recommended Threshold	Action
---	---	---
`OUT_OF_MEMORY`	1% of queries running out of direct memory	Add an engine and move workload
`RESOURCE ERROR`	1% of queries running out of heap memory	Add an engine and move workload
`ExecutionSetupException`	1% of queries exhibiting node disconnects	Add an engine and move workload
`ChannelClosedException (fabric server)`	1% of queries exhibiting node disconnects	Add an engine and move workload
`CONNECTION ERROR: Exceeded timeout`	1% of queries exhibiting node disconnects	Add an engine and move workload

Job State Durations

Use the job state durations (provided in milliseconds) in the `queries.json` to address SLA breaches.

Job State (`queries.json`)	Recommended Threshold	Action
---	---	---
Total Duration (*finish - start*)	p90 SLA aligns with your needs	Add all the states below
Pending (*pendingTime*)	p90 should not exceed 2000 milliseconds	Vertically scale the main coordinator node
Metadata Retrieval (*metadataRetrievalTime*)	p90 should not exceed 5000 milliseconds	Switch to a table format if the raw data is Parquet
Planning (*planningTime*)	p90 should not exceed 2000 milliseconds	Vertically scale the main coordinator node
Queued (*queuedTime*)	p90 should not exceed 2000 milliseconds	Add an engine and move workload
Execution Planning (*executionPlanningTime*)	p90 should not exceed 2000 milliseconds	Vertically scale the main coordinator node
Starting (*startingTime*)	p90 should not exceed 2000 milliseconds	Add an engine and move workload
Running (*runningTime*)	p90 SLA aligns with your needs	Add an engine and move workload

All italicized values can be found in `queries.json` as represented in the parentheses above.

Was this page helpful?

[Previous

Viewing Jobs](/current/admin/monitoring/jobs/)[Next

Exporting Logs](/current/admin/monitoring/exporting/)

Enabling Node Metrics

Available Prometheus Metrics

Parameters to Monitor for Scaling Capacity

Query Execution Errors

Job State Durations

Source:
<https://docs.dremio.com/current/admin/monitoring/exporting/>

Version: current [26.x]

Exporting Logs

You can use a monitor tool to collect logs from Dremio and forward them to a central location for analysis. See the steps for installing a monitoring tool that can be used to integrate with the following platforms:

[Amazon S3 and Azure Blob Storage](#)

[Datadog](#)

[Splunk](#)

Was this page helpful?

[Previous

Monitoring Dremio Nodes](/current/admin/monitoring/dremio-nodes)[Next

Amazon S3 and Azure Blob Storage](/current/admin/monitoring/exporting/aws-azure-storage)

Source: dremio-software-reference.md

Dremio Software - Reference

Source:

Version: current [26.x]

Reference

This section contains details about using Dremio's REST API, SQL commands and functions, and security advisories.

Admin CLI - The reference documentation on the `dremio-admin` CLI commands.

API Reference - The reference documentation on the Dremio REST APIs.

SQL Reference - The reference documentation on Dremio SQL to manage your data.

Security Bulletins - Security bulletins that disclose vulnerabilities found in Dremio's supported products.

Was this page helpful?

[Previous

API Reference](/current/reference/api/)[Next

Admin CLI](/current/reference/admin-cli/)

Source:
https://docs.dremio.com/current/reference/admin-cli/

Version: current [26.x]

On this page

Admin CLI

This topic summarizes the `dremio-admin` CLI commands.

Syntax

Syntax for dremio-admin commands

```
dremio-admin                [--config                <conf-dir>]
(encrypt|set-password|upgrade|recommend-reflections|delete-user-homespace|optimize-acls|
export-profiles|remove-duplicate-users|clean|remove-duplicate-roles|reset-catalog-search
|backup|delete-all-users|restore|repair-acls|nessie-maintenance) [args...]
```

Options

| Option | Description |

```
| --- | --- |
| `help, -h` | Displays usage information for the CLI commands. |
| `config <conf-dir>` | Used when the configuration file location is different than the
`/opt/dremio/conf` default directory. For example, if `dremio.conf`, `dremio-env`,
`logbook.xml`, and `logbook-admin.xml` are located in `/etc/dremio`, you will have to
run all of the `dremio-admin` commands as `./dremio-admin config /etc/dremio
<command> <arguments>`. |
```

Commands

Command	Description
---	---
`backup`	Backs up Dremio metadata and user-uploaded files.
`clean`	Cleans Dremio metadata.
`delete-all-users`	Deletes all internal Dremio users.
`delete-user-homespace`	Deletes the user's home space.
`encrypt`	Encrypt a user-supplied string.
`export-pats`	Exports personal access tokens (PATs).
`export-profiles`	Exports profiles of jobs from Dremio.
`import-pats`	Imports personal access tokens (PATs).
`nessie-maintenance`	Runs embedded Nessie repository maintenance tasks.
`optimize-acls`	Optimizes access control lists of sources, spaces, and datasets.
`recommend-reflections`	Recommend Reflections.
`remove-duplicate-roles`	Removes duplicate roles from Dremio.
`remove-duplicate-users`	Removes duplicate users from Dremio.
`reset-catalog-search`	Resets index to recover catalog search.
`repair-acls`	Repairs access control lists of sources, spaces, and datasets.
`restore`	Restores Dremio metadata and user-uploaded files.
`set-password`	Sets passwords for Dremio users (non-LDAP).
`upgrade`	Upgrades the KV store version. There are no options available for this command.

Log Directory

The default value for `DREMIO_ADMIN_LOG_DIR` is null (not set). When this parameter is *not* set, log files are *not* created.

To set the log directory, provide the log directory path by running the following:

Set log directory

```
export DREMIO_ADMIN_LOG_DIR=<path>
```

note

The export option must be set and access must be available for the user running the `dremio-admin` command.

Log Verbosity

Log verbosity is used in conjunction with `DREMIO_ADMIN_LOG_DIR`. Otherwise, all the

output is printed to `stdout`; there is no control on setting verbosity for `stdout`.

Verbosity options include:

TRACE

DEBUG

INFO (default)

WARN

ERROR

To set the log verbosity (default: INFO), provide the verbose level by running the following:

Set log verbosity

```
export DREMIO_ADMIN_LOG_VERBOSITY=<value>
```

For More Information

[Backup](#)

[Clean Metadata](#)

[Encrypt Credentials](#)

[Export and Import Personal Access Tokens \(PATs\)](#)

[Export Profiles](#)

[Perform Nessie Maintenance](#)

[Remove Duplicate Roles](#)

[Repair ACLs](#)

[Restore](#)

[Reset Password](#)

[Upgrade KV Store](#)

Was this page helpful?

[\[Previous](#)

[Dremio on Kubernetes\]\(/current/admin/admin-dremio-kubernetes/\)](#)[\[Next](#)

[Back up Dremio\]\(/current/reference/admin-cli/backup\)](#)

[Syntax](#)

[Options](#)

-

Commands

Log Directory

Log Verbosity

For More Information

Source:
<https://docs.dremio.com/current/reference/api/>

Version: current [26.x]

On this page

API Reference

The Dremio REST API is organized by resource types such as ``sources`` and is designed around RESTful principles.

``GET`` is used to retrieve existing resources

``POST`` creates new resources

``PUT`` updates resources

``DELETE`` removes resources

Base URL

All API URLs referenced in this documentation have the following base URL unless otherwise specified:

Base URL

```
{hostname}/api/v3
```

Versions prior to ``v3`` are considered internal and are subject to change without notice.

In this documentation, curly braces (``{}``) are used to indicate sections of URLs where you have to supply a value. For example:

User-supplied values in URLs

```
/api/v3/source/{id}
```

Authentication

Each REST request requires an authorization header with a Dremio access token to authenticate the requester unless otherwise indicated. Dremio accepts three types of access tokens for authenticating REST requests.

OAuth access tokens are created in Dremio using the Dremio REST API.

Personal access tokens are created in the Dremio console or with REST.

Authentication tokens are generated from a username and password using the Dremio v2 API.

All Dremio access tokens are Bearer tokens and can be used in the REST authorization header of each REST request.

Example Dremio REST request

```
curl -X GET 'https://{hostname}/api/v3/{path_to_endpoint}' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

OAuth Access Tokens Enterprise

Users can create OAuth access tokens by exchanging a local or LDAP username and password, a PAT, or an external JWT using the `/oauth/token` REST API. Dremio provides [sample code](#) for each of these cases.

Dremio recommends OAuth access tokens to improve system security:

OAuth access tokens are typically short-lived, reducing the window of opportunity for attackers if a token is compromised.

Users must manually revoke compromised or suspected PATs, often leading to forgotten, unused tokens.

Personal Access Tokens Enterprise

Any user can create personal access tokens (PATs) [in the Dremio console](#) or [using REST](#). Users can configure the lifetime of each personal access token, from 1 day to a maximum defined by the `auth.personal-access-token.max_lifetime_days` [support setting](#).

Dremio Authentication Tokens

Users can generate authentication tokens from their Dremio username and password. Authentication tokens have a nonconfigurable lifetime of 30 hours.

caution

Generating an authentication token requires API v2. API versions prior to v3 are considered internal and are subject to change without notice.

To generate an authentication token:

Send an API request to the login URL with your Dremio username and password in the request body.

Example

Request

```
curl -X POST 'http://{hostname}/apiv2/login' \  
--header 'Content-Type: application/json' \  
--data-raw '{  
  "userName": "dremio",  
  "password": "dremio123"  
}'
```

Response

```
{  
  "token": "4ksrt534vk7fkq64xh55g7776b",  
  "userName": "dremio",  
  "firstName": "Dre",  
  "lastName": "Mio",  
  "expires": 1686578200000,  
  "email": "dremio@dremio.test",  
  "userId": "5a679dd5-52d7-402a-871d-7fbee3fe8007",  
  "admin": true,  
  "clusterId": "7468ce46-58af-4dce-a42f-4c51048968f5",  
  "clusterCreatedAt": 1681311939728,  
  "version": "24.0.0-main-202305040803350903-2d5579e3",  
  "permissions": {  
    "canUploadProfiles": true,  
    "canDownloadProfiles": true,  
    "canEmailForSupport": true,  
    "canChatForSupport": false,  
    "canViewAllJobs": true,  
    "canCreateUser": true,  
    "canCreateRole": true,  
    "canCreateSource": true,  
    "canUploadFile": true,  
    "canManageNodeActivity": true,  
    "canManageEngines": true,  
    "canManageQueues": true,  
    "canManageEngineRouting": true,  
    "canManageSupportSettings": true  
  },  
  "userCreatedAt": 1681311939789  
}
```

note

If your password includes single or double quotes, you may need to escape the quotes in your authentication token request. The required escapes vary depending on how you send the request. For example, if you use cURL and the password is `example'6852"`, the password value should be `example\'"6852\"` in the authentication token request.

Use the `token` attribute of the JSON return object as a Bearer token, or append the

prefix ``_dremio`` to compose a self-contained token as ``_dremio<tokenstring>``.

Example: Request Using A Self-Contained Authorization Token

```
curl -X GET 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: _dremio4ksrt534vk7fkq64xh55g7776b' \
--header 'Content-Type: application/json'
```

Errors

Error messages will be sent back in the response body using the following format:

Error Messages Format

```
{
  "errorMessage": "brief error message",
  "moreInfo": "detailed error message"
}
```

Query Parameters

Dremio supports query parameters for many API endpoints. The documentation for each API lists the supported query parameters for specific endpoints, along with any default and maximum values for the query parameters for that endpoint.

pageToken Query Parameter

Use the ``pageToken`` query parameter to split large sets of results into multiple pages.

Endpoints may support the ``pageToken`` query parameter based on either a built-in maximum number of results per page or a user-specified maximum that is established with a separate query parameter. The documentation for each API lists the built-in maximum or the query parameter to use to specify a maximum, as applicable.

note

Do not change any other query parameters included in the request URL when you use ``pageToken``.

Built-in Maximum

If the endpoint has a built-in maximum number of results per page, responses automatically include a page token attribute when the response contains more results than the built-in maximum. Use the value for this token in the request URL as the ``pageToken`` value to retrieve the next page of results.

As an example, the Reflection summary endpoint supports the ``pageToken`` parameter. If the Reflection summary contains more than 50 results, the response will include the ``nextPageToken`` attribute. To retrieve the next 50 results, add ``?pageToken={nextPageToken_value}`` to the request URL:

Example Request with pageToken Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?pageToken=BhQxNjc0MjhlYi030TM2LTRlYtYTFmYi0yM2IxYWM2ZTk0NTQSA== ' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

For subsequent requests, replace the `pageToken` value in the request URL with the token value from the previous response. If the response does not include a token attribute and value, you have retrieved the last page of available results.

User-Specified Maximum

For endpoints that require users to specify a maximum number of results per page with a separate query parameter, responses only include a page token attribute if your initial request URL includes the separate query parameter and the response contains more results than the maximum you specify. Add the value for this token to the request URL as the `pageToken` value, keeping the separate query parameter as well, to retrieve the next page of results.

Catalog API endpoints for retrieving non-filesystem [sources](#), [spaces](#), and [folders](#) by ID or path support the [`maxChildren` query parameter](#) for specifying the maximum number of child objects to include in each response. If the response contains more than the specified number of child objects, the response includes the `nextPageToken` attribute. To retrieve the next page of results, add `?pageToken={nextPageToken_value}` to the request URL. This example shows a request URL that uses the `nextPageToken` query parameter with the `maxChildren` query parameter set to 25:

Example Request with maxChildren and pageToken Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4?maxChildren=25&pageToken=BhQxNjc0MjhlYi030TM2LTRlYtYTFmYi0yM2IxYWM2ZTk0NTQSA== ' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

For subsequent requests, replace the `pageToken` value in the request URL with the token value from the previous response. If the response does not include a token attribute and value, you have retrieved the last page of available results.

maxResults Query Parameter

Use the `maxResults` query parameter to specify the maximum number of results to retrieve in each request.

For example, if you want to retrieve no more than 25 results for an endpoint that supports the `maxResults` query parameter, append `?maxResults=25` to the request URL:

Example Request with maxResults Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?maxResults=25' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

filter Query Parameter

Use the ``filter`` query parameter to filter responses so that they include only results with the specified attributes and values. The value for the ``filter`` query parameter is a URL-encoded JSON string that represents a JSON object that specifies the desired attributes and values.

As an example, the [Reflection summary](#) endpoint supports the ``filter`` query parameter for certain specific attributes. To retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with ``samples.dremio.com`` in their paths, the ``filter`` JSON object would look like this:

Example JSON Object for Filter

```
{
  "reflectionType": ["RAW"],
  "refreshStatus": ["MANUAL", "SCHEDULED"],
  "enabledFlag": true,
  "reflectionNameOrDatasetPath": "samples.dremio.com"
}
```

To use the JSON object as the ``filter`` value, convert it to URL-encoded JSON and add it to the request URL:

Example Request with filter Query Parameter

```
curl -X GET
'https://{hostname}/api/v3/reflection-summary?filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the endpoint-specific documentation to learn which attributes each endpoint supports for the ``filter`` query parameter.

orderBy Query Parameter

Use the ``orderBy`` query parameter to organize the response in ascending or descending order based on the value of the specified attribute. The default is ascending order. To specify descending order, add a ``-`` character before the attribute name.

For example, the [Reflection summary](#) endpoint supports ordering the response by `ReflectionName`, `datasetName`, or `reflectionType`. To organize the response in ascending order by `ReflectionName`:

Example Request with orderBy Query Parameter (Ascending Order)

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

To organize the response in descending order, add a ``-`` before the attribute name:

Example Request with orderBy Query Parameter (Descending Order)

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=-reflectionName' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Read the endpoint-specific documentation to learn which attributes each endpoint supports for the ``orderBy`` query parameter.

limit and offset Query Parameters

The ``limit`` query parameter allows you to retrieve a specific number of results. For endpoints that support the ``limit`` query parameter, you can specify the number of results to retrieve. For example, if you only want to retrieve the first 10 available results, add ``?limit=10`` to the request URL:

Example Request for First 10 Results with Limit Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=10' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

The ``offset`` query parameter allows you to skip a specific number of results in the response. When a response contains many results, you can use the ``limit`` and ``offset`` query parameters together to break the response into pages.

For example, consider a job result response object that contains 5000 results. The Job API allows you to retrieve a maximum of 500 results per request. To retrieve all 5000 results, start by adding ``?limit=500`` to the request URL to retrieve the first 500:

Example Request for First 500 Results with Limit Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

In the next request, to retrieve the next 500 results (rows 501-1000), add ``&offset=500`` to the request URL:

Example Request for Results 501-1000 with limit and offset Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500&offset=500' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

To retrieve the next 500 results (rows 1001-1500), increment the `offset` parameter to 1000 in the next request:

Example Request for Results 1001-1500 with limit and offset Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500&offset=1000' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Continue incrementing the `offset` parameter in requests until you have retrieved all 5000 results.

Read the documentation for each API to learn about endpoint-specific support for the `limit` and `offset` query parameters.

type Query Parameter

Use the `type` query parameter to limit your request so that the response includes only results for the type you specify.

For example, if an endpoint supports the `type` query parameter, and the endpoint's list of valid values includes `SOURCE`, you can limit the response so that it includes only results for sources. Append `?type=SOURCE` to the request URL:

Example Request with type Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/privileges?type=SOURCE' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `type` query parameter, including lists of valid values.

include and exclude Query Parameters

Some APIs exclude non-default attributes or include lengthy attributes in the default GET responses. These APIs support the `include` and `exclude` query parameters, which you can use to include or exclude certain attributes in the responses for GET endpoints.

The `include` query parameter allows you to include non-default attributes in the response. For example, in the Catalog API, you can include a catalog object's `permissions` array in the response:

Example Request with include Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/ffbe8c1d-1db7-48d1-9c58-f452838fedc0?include=permissions' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```


The ``exclude`` query parameter allows you to exclude supported attributes from the response. For example, this Catalog API request excludes the object's ``children`` attribute from the response:

Example Request with exclude Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/ffbe8c1d-1db7-48d1-9c58-f452838fedc0?exclude=children' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the ``include`` and ``exclude`` query parameters, including information about supported values.

createdBy Query Parameter

Use the ``createdBy`` query parameter to limit the request to objects created by a specific user.

For example, this Scripts API request retrieves only scripts that were created by the user whose ID is ``8be516f3-04c4-4d19-824d-5a70b3c4442e``:

Example Request with createdBy Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?createdBy=8be516f3-04c4-4d19-824d-5a70b3c4442e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

ownedBy Query Parameter

Use the ``ownedBy`` query parameter to limit the request to objects owned by a specific user.

For example, this Scripts API request retrieves only scripts that are owned by the user whose ID is ``8be516f3-04c4-4d19-824d-5a70b3c4442e``:

Example Request with ownedBy Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?ownedBy=8be516f3-04c4-4d19-824d-5a70b3c4442e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

search Query Parameter

Use the ``search`` query parameter to limit the request to objects that contain values that include the search string.

For example, the Scripts API supports the `search` query parameter for the name attribute. This Scripts API request retrieves only scripts whose values for the name attribute include `dev`:

Example Request with search Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?search=dev' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `search` query parameter, including information about searchable attributes.

maxChildren Query Parameter

The `maxChildren` query parameter allows you to specify the maximum number of child objects to include in each response. This example shows a request URL that uses the `nextPageToken` query parameter with the `maxChildren` query parameter set to 25:

Example Request with maxChildren Query Parameter

```
curl -X GET
'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4?maxChildren=25'
\
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Use the `maxChildren` query parameter in concert with the pageToken` query parameter to split large sets of results into multiple pages.

Response Headers

Dremio API responses include HTTP headers that provide additional information about responses. Each header includes a case-insensitive name and a value, separated by a colon.

The following response headers are common to Dremio API endpoints:

Header Name	Example Value	Description
Allow	GET,OPTIONS	Request methods the endpoint supports.
Cache-Control	no-cache, no-store	Caching instructions for browsers and shared caches.
Content-Length	2188	Size of the response body, in bytes.
Content-Type	application/json	MIME type of the object.
Date	Fri, 14 Apr 2023 19:39:53 GMT	Date and time when the response originated.
Vary	Accept-Encoding, User-Agent	Names of request headers that could have affected the response's generation.
x-content-type-options	nosniff	Instructions about following the MIME type in the content-type header. Blocks content sniffing.
x-xss-protection	1; mode=block	Instructions used to stop pages from loading when

a browser detects reflected cross-site scripting attacks. |

Was this page helpful?

[Previous

Admin CLI](/current/reference/admin-cli/)[Next

Catalog](/current/reference/api/catalog/)

Base URL

Authentication

OAuth Access Tokens Enterprise

Personal Access Tokens Enterprise

Dremio Authentication Tokens

Example

Errors

Query Parameters

pageToken Query Parameter

maxResults Query Parameter

filter Query Parameter

orderBy Query Parameter

limit and offset Query Parameters

type Query Parameter

include and exclude Query Parameters

createdBy Query Parameter

ownedBy Query Parameter

search Query Parameter

maxChildren Query Parameter

Response Headers

Source:
<https://docs.dremio.com/current/reference/sql/>

Version: current [26.x]

On this page

SQL Reference

Dremio provides comprehensive SQL access to your data, no matter where it is stored.

[Data Types](#)

[SQL Functions](#)

[SQL Commands](#)

[Reserved Words](#)

[System Tables](#)

[Table Functions](#)

[Information Schema](#)

Additional Resources

Find out more about using SQL by enrolling in the [SQL for Data Analysts course in Dremio University](#).

Was this page helpful?

[Previous

API Reference](/current/reference/api/)[Next

Data Types](/current/reference/sql/data-types/)

Additional Resources

Source:
<https://docs.dremio.com/current/reference/bulletins/>

Version: current [26.x]

Security Bulletins

Dremio publishes security bulletins that disclose vulnerabilities found in our supported products to inform customers about risks that may be present in their production environments.

Security bulletins are usually published when fixes are available in the affected products. In some cases, we may disclose a vulnerability before the fix is available.

Security bulletins include the following information:

Type

Qualitative rating as determined by CVSSv3.1 analysis

Issue description

Issue impact

Available mitigations or fixes

Bulletin	Type	CVSS Rating	Subject	Description
---	---	---	---	---
2025-04-21-01	Vulnerability	High	Security Update	An authenticated API endpoint allows arbitrary file deletion.
2024-02-07-01	Vulnerability	Medium	Security Update	The COPY INTO command does not verify users' SELECT privileges.
2024-01-12-01	Vulnerability	High	Security Update	Path traversal vulnerability bypassed folder-level role-based access control (RBAC).
2024-01-09-01	Vulnerability	High	Security Update	The Dremio-to-Dremio connector does not fully validate table-level access in certain cases.
2023-07-22-03	Vulnerability	Medium	Security Update	Potential unintended user access to restricted data as a result of previously cached view.
2023-07-22-02	Vulnerability	Medium	Security Update	Potential unintended user access to restricted data as a result of accelerated DML operation.
2023-07-22-01	Vulnerability	Medium	Security Update	Potential unintended user access to restricted data as a result of previously-executed cached plans.

Was this page helpful?

[\[Previous](#)

[SQL Reference\]\(/current/reference/sql/\)\[Next](#)

[2025-04-21-01\]\(/current/reference/bulletins/2025-04-21-01\)](#)

Source: [dremio-software-security.md](#)

Dremio Software - Security

Source:
<https://docs.dremio.com/current/security/>

Version: current [26.x]

Security and Compliance Enterprise

Dremio offers extensive security measures to help protect the integrity of your data, such as access control and federation with external identity providers. Dremio provides flexible native security features and integration with a wide range of third-party tools so that your organization can adhere to compliance and regulatory standards, enforce fine-grained permissions for your users, and retain your existing tools for authentication and authorization.

Authentication – Manage user identities, authentication, and enterprise identity providers.

Access Control – Grant fine-grained permissions at the user or role level using native Dremio features or an integrated external data security product.

Integrations – Manage access control and data governance through third-party

integrations.

Secrets Management - Use a secrets management service to provide sensitive information like passwords and secret access keys instead of providing it in plaintext.

Audit Logging - Ensure the integrity of your network using virtual network (VNet) configurations for outbound connections from Dremio executors.

Compliance - Review the current compliance measures and audits Dremio has undergone to ensure top-level security for your data.

As the Dremio Shared Responsibility Models outline, platform security is a shared responsibility between Dremio and you. The Shared Responsibility Models lay out Dremio's responsibilities for providing vulnerability management and application security and your responsibilities for deployment and operations.

To configure external services, read:

Integrate with Lake Formation

Integrate with Privacera

Apache Ranger Row-Level Filtering & Column-Masking

LDAP

Was this page helpful?

[Previous

Use Reflections in Nessie Source
Branches](/current/acceleration/manual-reflections/using-nessie-branches-with-reflections)[Next

Authentication](/current/security/authentication/)

Source:
<https://docs.dremio.com/current/security/authentication/>

Version: current [26.x]

On this page

Authentication

Dremio supports several types of authentication for identity providers, client connections, and user types, including both regular users and service users.

Authentication Methods by Application Type

App Type	Enterprise OIDC Provider	LDAP or Dremio Local Provider
---	---	---
Dremio Console	* Single Sign-On * Personal Access Token	* Username and

Password * Personal Access Token |

| **User Clients & Applications** | * Personal Access Token * **OAuth-based authentication**: External JWT Exchange or External JWT with Legacy JDBC | * Personal Access Token * Username and Password with JDBC/ODBC |
| **M2M Applications** | * Service Users with External Service Principals | * OAuth Client Credentials * **Legacy Migration**: PAT Exchange or Obtaining OAuth Tokens with Username and Password |

Dremio Console Authentication Methods

Single Sign-On

The user is authenticated by the configured OIDC identity provider, including automatic authentication if the user is already signed in to the identity provider.

Username and Password

The user provides a username and password combination for authentication. See [User Management](#) for information on adding and managing local and external users.

Personal Access Token

A personal access token (PAT) is used in place of a user password. PATs provide a convenient way to create a client connection without exposing a user's password, but can pose a security risk if not properly managed. PATs can be configured with long lifetimes, and lost or compromised tokens may allow access to sensitive data until the token expires. Before use, the administrator must [activate PATs](#) for the Dremio cluster.

To use a PAT, the user must follow these steps:

[Create a PAT](#) in the Dremio console. Users can create additional PATs using the Dremio console or the [PAT creation](#) REST API.

Use the PAT to connect with the Dremio console, [Arrow Flight SQL JDBC](#), [Arrow Flight SQL ODBC](#), [Legacy JDBC](#), or [Dremio REST](#).

User Applications Authentication Methods

External JWT

Client apps can request OAuth 2.0 JSON Web Tokens (JWTs) from external token providers, allowing users to authenticate through custom or third-party applications without exposing their credentials to the client application.

After obtaining an external JWT, the client app can create connections to Dremio using the [Legacy JDBC driver](#). However, Dremio recommends external JWT token exchange because Dremio OAuth access tokens are smaller and verification is faster.

To use an external JWT, the administrator must configure Dremio to use the Enterprise OIDC provider as an [external token provider](#).

After configuration, a client application performs the following steps:

A user authenticates with the external token provider and the client receives a JWT.

Create a connection to Dremio using the Legacy JDBC and the external JWT.

External JWT Exchange

Exchanging the external JWT for an OAuth access token enables additional connection choices after authenticating with the external token provider. A client application performs the following steps:

A user authenticates with the external token provider and the client receives a JWT.

Use the `/oauth/token` REST API to exchange the JWT for an OAuth access token.

Create a connection to Dremio using Arrow Flight SQL JDBC, Arrow Flight SQL ODBC, Legacy JDBC, or Dremio REST and the OAuth access token.

M2M Applications Authentication Methods

OAuth Client Credentials

Service users authenticate using the OAuth 2.0 client credentials flow, where a client ID and client secret are exchanged for access tokens. This is the primary authentication method for service users and provides:

Automated authentication without manual login processes

Short-lived access tokens that enhance security

Centralized credential management through the OAuth system

Audit trails for programmatic access

To use OAuth client credentials:

Create a service user in the Dremio console under **Settings > User Management > Service Users**. Upon creation, Dremio generates a unique client ID and client secret.

Use the `/oauth/token` REST API to exchange the client ID and client secret for an OAuth access token.

Create a connection to Dremio using Arrow Flight SQL JDBC, Arrow Flight SQL ODBC, Legacy JDBC, or Dremio REST and the OAuth access token.

External Service Principal Authentication

You can configure Dremio service users to authenticate using service principals from Microsoft Entra ID or another OIDC provider. This allows service users to authenticate using JWTs from external identity providers, which are then exchanged for Dremio OAuth access tokens.

This method is useful for organizations that want to:

Centralize service principal management in their identity provider

Use existing Microsoft Entra ID service principals for Dremio access

Maintain consistent authentication patterns across multiple systems

To use external service principal authentication:

Create a service user in the Dremio console and configure external credentials linking to your service principal in Microsoft Entra ID or another OIDC provider.

The service user authenticates with the external identity provider and receives a JWT.

Use the `/oauth/token` REST API to exchange the external JWT for an OAuth access token.

Create a connection to Dremio using Arrow Flight SQL JDBC, Arrow Flight SQL ODBC, Legacy JDBC, or Dremio REST and the OAuth access token.

Legacy Authentication Methods

Obtain OAuth Access Tokens with Username and Password

Organizations transitioning to OAuth-based authentication can use a username and password from a traditional user account to obtain an OAuth access token. This method allows teams to implement OAuth-based authentication immediately while planning their migration to dedicated service users and any associated configuration of an external identity provider.

Users follow these steps to exchange a username and password:

Use the `/oauth/token` REST API to obtain OAuth access tokens using a username and password.

Create a connection to Dremio using Arrow Flight SQL JDBC, Arrow Flight SQL ODBC, Legacy JDBC, or Dremio REST and the OAuth access token.

Use the optional refresh token to create OAuth access tokens to obtain fresh OAuth access tokens as they expire.

PAT Exchange

PAT Exchange serves as a migration bridge for existing applications that currently use PATs but need to integrate with systems expecting OAuth access tokens. This method allows organizations to maintain existing PAT-based workflows while transitioning to proper service user authentication.

Users follow these steps to exchange a PAT:

Create a PAT in the Dremio console or using the REST API after creating the first token.

Use the `/oauth/token` REST API to exchange the PAT for an OAuth access token.

Create a connection to Dremio using [Arrow Flight SQL JDBC](#), [Arrow Flight SQL ODBC](#), [Legacy JDBC](#), or [Dremio REST](#) and the OAuth access token.

Dremio Authentication Token

Dremio authentication tokens are generated from your Dremio username and password. This authentication method uses the prior generation `/apiv2/login`` endpoint, now internal and subject to change without notice. See [Dremio Authentication Tokens](#) for additional information.

Was this page helpful?

[\[Previous](#)

[Security and Compliance\]\(/current/security/\)\[Next](#)

[Manage Users\]\(/current/security/authentication/users\)](#)

[Authentication Methods by Application Type](#)

[Dremio Console Authentication Methods](#)

[Single Sign-On](#)

[Username and Password](#)

[Personal Access Token](#)

[User Applications Authentication Methods](#)

[External JWT](#)

[External JWT Exchange](#)

[M2M Applications Authentication Methods](#)

[OAuth Client Credentials](#)

[External Service Principal Authentication](#)

[Legacy Authentication Methods](#)

[Obtain OAuth Access Tokens with Username and Password](#)

[PAT Exchange](#)

[Dremio Authentication Token](#)

Source:
<https://docs.dremio.com/current/security/rbac/>

Version: current [26.x]

On this page

Access Control Enterprise

Dremio allows for the implementation of granular-level privileges, which defines a user/role's access privilege and available actions for specific objects, such as folders and datasets. This is called access management, and gives administrators the ability to restrict access to any object in Dremio.

Privileges – Privileges enable users to perform explicit operations on objects in Dremio. Additionally, privileges may be set on individual datasets (tables or views) or whole schemas, allowing for a simplified configuration with larger catalogs.

Row-access and column-masking policies – Row-access and column-masking policies enable you to grant users access to particular rows or columns.

The following APIs are available for managing access control:

Grants APIs

Privileges API

Roles API

Users API

A wide range of SQL commands are also available:

Privileges SQL Commands

Roles SQL Commands

Users SQL Commands

Object Hierarchy

Each object resides within a container in a hierarchy of containers. The upper-most container exists as the system user, or administrator account. All other objects are contained within sources or spaces, organized into folders. The hierarchy of these objects is illustrated below.

![[/assets/images/rbac-object-hierarchy-a829e25a2980a7f00d7cc2a85ccbbf00.png)]

Inheritance

The objects to which privileges are granted depend on the inheritance model. In other words, granting access to a parent object, such as a folder, also gives that user access to any existing and future datasets contained in that folder. For example, giving a user privileges to ALL DATASETS will only grant the user access to existing datasets, not the folders that contain the datasets. In comparison, granting privileges at the source level will extend that user's access to the source's existing and future folders/schema and datasets. The object to which a user's privileges are applied is also known as the scope, and follow a parent-child relationship.

By the rules of inheritance, user or group access may be granted as high or low in the object hierarchy as you wish for access to reach.

Permissions granted to an individual table or view mean that a user's access only extends to that dataset, not to the parent folder or other datasets created in the same

folder. So if a user only needs access to a single dataset, administrators need only grant privileges to that object.

Consider the image above, which shows an example of object structure in Dremio. If a user is granted privileges to a single dataset, such as ``TableA1``, then that is the one object they have access to. However, if a user is granted privileges at the folder level, such as ``Folder1``, then that user's access extends to any existing and future child objects created, including ``FolderA``, ``TableA1``, ``TableB1``, and so on.

note

If a user has privileges for a single table, they may create views based on that dataset, but with the user now having ``ALTER`` and ``MANAGE GRANTS`` privileges for any view. However, the user still retains the same privileges as before with the original dataset. For more information, read [View Delegation](#).

Scope

Scope is a concept used to describe what objects a user or group has access to. Privileges are assigned by object, which ultimately determines what a grantee may perform set functions upon. For example, you may set a user's scope to ``FolderA``, which will give the user access to all existing and future datasets contained in the folder, as well as the datasets' wikis. But they will not have access to any other folders or the source. The object a user is granted access to is dependent on the inheritance model, which means based on the object type, it may contain child objects. For example, if a user is granted privilege to a folder, the user's access also extends to all existing and future datasets contained in that folder.

For example, ``user1`` is granted the ``SELECT`` privilege to the folder ``FolderC``. This object contains multiple datasets, which the user may now access. However, there exists a parent folder and another subfolder with its own datasets.

Because of the established scope, ``user1`` may not access ``FolderD`` because they were only granted access to ``FolderC``'s objects.

Current vs. Future Objects

Based on the selected scope, you may restrict a user's access to future and existing datasets. For example, if you select a single table as the scope of a user's privilege, then that user may only perform that action to the existing dataset, as well as any future views they create using that table. However, they may not access any views created from a table by another user (see the example below). However, if the scope is instead set at the folder level, then the user may perform the granted privilege to all tables and views contained in that folder (see the example below).

Ownership

Object ownership is a security feature used to control access to an object. In Dremio, each object must have an owner, and may have only one owner. Ownership is

automatically granted to the user who initially created the object. For example, when `user1` creates an S3 data source, Dremio automatically assigns ownership of the source to `user1`.

The privileges included in object ownership depend on your configuration.

By default, ownership includes all privileges for that object. The object owner can grant or revoke access privileges to the object and its child objects, modify an object's settings, and delete the object as desired. See [Granting Privileges Using SQL Commands](#) for more information.

Managed access spaces centralize the administration of access privileges in shared spaces to a limited set of users and roles, including the space owner. By limiting privilege grant authority, managed access spaces help ensure consistent and controlled access policies and reduce the risk of unauthorized access. See [Managed Access Spaces](#).

The following behaviors and limitations apply to ownership:

Each object may only have one owner.

An object's creator is automatically granted ownership.

Object ownership may be assigned or modified to a new user or role with the `GRANT OWNERSHIP` command.

The object's access control settings may not work if the owner is deleted or removed. See [View Delegation](#).

Object owners may be identified by querying the `sys."tables"` table or `sys.views` table. If an object has no owner, the `owner_id` will display as `$unowned`.


Managed Access Spaces

Managed access spaces centralize the administration of access privileges to a limited set of users and roles, including:

Owner of the space

Dremio administrator

Users or roles explicitly granted the `MANAGE GRANTS` privilege on the object or any of its parents

When using managed access spaces, Dremio displays shared spaces with a lock icon  on the [Datasets](#) and [SQL Runner](#) pages. Owners of folders, views, and functions in a locked space cannot grant or revoke privileges on those objects to other users or roles.

User or role	Grants/revokes privileges in default spaces	Grants/revokes privileges in managed access spaces
---	---	---
Dremio administrator	Yes	Yes
Owner of a shared space	Yes	Yes
Owner of a folder in a shared space	Yes	No
Owner of a view in a shared space	Yes	No

| Owner of a function in a shared space | Yes | No |
| User or role with `MANAGE GRANTS` | Yes | Yes |

Managed access spaces do not impact:

Any user home space

Sources, including Nessie catalogs

Global objects, such as scripts and user-defined functions

Managed access spaces do not override a `MANAGE GRANT` privilege granted at system scope.

The Dremio administrator can activate managed access spaces by setting the ``security.access-control.managed-access-spaces.enabled`` support key on the Support Settings page.

View Delegation

View delegation means that the data in tables with restricted access may be available to other Dremio users by creating views. View delegation is the critical capability of the Dremio semantic layer that allows users to run queries without accessing the underlying tables and views directly.

The fundamental principles of view delegation include the following:

The privileges of a view's owner determine whether the view can use dependent tables and views.

Additional user access to a view is controlled by privilege grants directly on the view, forming a privilege chain from the view to the underlying table.

note

A shared view selects from the underlying dataset using the view owner's permissions at the time of the view's last modification, even if the end user querying the view lacks privileges to modify the underlying table. This applies to each table on the data graph and chain of datasets.

View delegation is different from privilege assignment. View delegation is an implicit delegation of the `SELECT` privilege on underlying objects, which means that users who run queries on a view must have access privileges on the view but do not need privileges on underlying tables. Privilege assignment is an explicit delegation providing direct access to an object.

Example 1: View Delegation

``user1`` has the `SELECT` privilege on ``table1`` and creates ``view1`` to filter and transform data in ``table1``. ``user2`` asks for access privileges to run queries on ``view1`` as well. ``user2`` may obtain the `SELECT` privilege for ``view1`` from the following authorized users:

By default, view owners such as ``user1`` can grant and revoke privileges to other users, as appropriate.

A limited set of users and roles, such as the space owner, can grant or revoke privileges in managed access spaces.

Dremio administrators or other users with the `MANAGE GRANTS` privilege can grant privileges to other users.

If access for ``user2`` is appropriate, the authorized user runs ``GRANT SELECT ON VIEW view1 TO USER user2`` to grant the `SELECT` privilege to ``user2``. After ``user2`` obtains the `SELECT` privilege, they can run queries on ``view1``, utilizing the privilege of ``user1`` as owner to ``view1`` to `SELECT` from ``table1``.

Object	<code>`user1`</code>		<code>`user2`</code>	
---	---		---	
<code>`view1`</code>	OWNERSHIP		SELECT	
<code>`table1`</code>	SELECT		None	

Privileges by user in Example 1

The following table describes the actions that each user may perform based on their privileges:

Task	Works for <code>`user1`</code>		Works for <code>`user2`</code>	
---	---		---	
Use <code>`view1`</code> in queries	Yes, <code>`user1`</code> owns <code>`view1`</code> and has the <code>SELECT</code> privilege on <code>`table1`</code> . Yes, <code>`user2`</code> has the <code>SELECT</code> privilege on <code>`view1`</code> and the owner of <code>`view1`</code> has the <code>SELECT</code> privilege on <code>`table1`</code> .			
Modify the query in <code>`view1`</code>	Yes, <code>`user1`</code> is the owner of <code>`view1`</code> . Ownership includes the <code>ALTER</code> privilege to modify the view definition. No, <code>`user2`</code> is not the owner of <code>`view1`</code> and does not have the <code>ALTER</code> privilege.			
Use <code>`table1`</code> in queries	Yes, <code>`user1`</code> has the <code>SELECT</code> privilege on <code>`table1`</code> . No, <code>`user2`</code> has no privileges on <code>`table1`</code> and cannot see it in the Dremio user interface.			

Tasks by user in Example 1

Example 2: View Delegation with Revoked Access to the Original Table

To continue the previous example, ``user1`` has `SELECT` access to ``table1``, which gives ``user1`` access through ``view1``. An administrator revokes the `SELECT` access of ``user1`` on ``table1``.

Object	<code>`user1`</code>		<code>`user2`</code>	
---	---		---	
<code>`view1`</code>	OWNERSHIP		SELECT	
<code>`table1`</code>	None		None	

Privileges by user in Example 2

The following table describes the actions that each user may perform based on their privileges:

Task	Works for <code>`user1`</code>		Works for <code>`user2`</code>	
---	---		---	
Use <code>`view1`</code> in queries	No, <code>`user1`</code> no longer has <code>SELECT</code> on the underlying <code>`table1`</code> .			
No, <code>`user2`</code> no longer has a chain of permission through <code>`user1`</code> to <code>`table1`</code> .				

| Modify the query in `view1` | No, `user1` is the owner of `view1` but any attempts to edit `view1` will fail unless the references to `table1` are removed since `user1` can no longer access `table1`. | No, `user2` is not the owner of `view1` and does not possess the ALTER privilege. |
| Use `table1` in queries | No, `user1` has no privileges on `table1`. | No, `user2` has no privileges on `table1`. |

Tasks by user in Example 2

Privileges

Privileges refer to the defined levels of access or permissions that are assigned to roles or users within Dremio. Privileges determine the operations a user or role can perform on securable objects. Examples of privileges in Dremio include SELECT on a table or view, INSERT on a table, DELETE on a table, CREATE TABLE on a folder, and MANAGE GRANTS on any object.

The assignment of privileges to roles, or users, should be based on the principle of least privilege, where users or roles are given only the minimum privileges required to perform their tasks effectively.



Privileges can be managed using SQL, APIs, or the Dremio Console.

For more information, please refer to [Privileges](#).

Granting Privileges Using the Dremio Console

You can share catalog objects with others in your organization by granting privileges on the objects to users and roles as follows:

Locate the desired object.

Click  This is the icon that represents more actions. or  This is the icon that represents more actions. depending on the object.

In the object settings dialog, select **Privileges** from the settings sidebar.

note

For some object types, the settings dialog automatically opens to display the privilege settings, and you do not need to select the Privileges tab.

In the Privileges dialog, in the field under **Add User/Role**, enter the exact names of the users and roles to which you want to grant privileges.

note

Because all users are members of the PUBLIC role, you can use the PUBLIC role to grant privileges to all users.

Click **Add to Privileges**.

For each entry in the **Add User/Role** field that matches a user or role in Dremio, a record appears in the USERS/ROLES table.

-

In the USERS/ROLES table, toggle the checkbox for each privilege you want to grant for that user or role. For a description of the privilege, hover over the column name in the USERS/ROLES table. See the example below:

(Optional) Repeat steps 4-6 if you want to add more users or roles and grant them privileges.

When finished, click **Save**.

Revoking Privileges Using the Dremio Console

To revoke user and role privileges, complete the following steps:

Locate the desired object.

Click !This is the icon that represents more actions. or !This is the icon that represents more actions. depending on the object.

In the object settings dialog, select **Privileges** from the settings sidebar.

note

For some object types, the settings dialog automatically opens to display the privilege settings, and you do not need to select the Privileges tab.

In the USERS/ROLES table, locate the desired user or role record. If the user or role is not listed, then they do not have specific privileges on the object.

To revoke some but not all privileges for the user or role, clear the checkboxes in the columns for the privileges you wish to revoke.

To revoke all privileges for a user or role, click !This is the icon that represents more actions. next to the user or role name and select **Remove**.

For a description of the privilege, hover over the column name in the USERS/ROLES table. See the example below:

When finished, click **Save**.

note

If a user has a specific privilege on an object through their memberships in multiple roles and the privilege is revoked for one of the roles, the user retains the privilege until it is revoked on the same object for all roles to which the user belongs.

tip

You can also grant or revoke privileges using SQL commands or APIs.

Granting Privileges Using SQL Commands

When granting privileges to users and roles with SQL commands, you may follow one of

three methods: granting to a single dataset, granting to ALL DATASETS, and granting to a scope. Examples of these methods may be found under each section.

Each example includes an SQL command. For more information about command syntax, review the [Privileges \(GRANT/REVOKE\) SQL commands](#).

note

Because all users are members of the PUBLIC role, you can use the PUBLIC role to grant privileges to all users.

Granting to a Single Dataset

When you have a user that needs access to only one table and no other objects, then you would simply assign them privileges for that dataset (see the example scenario outlined below).

You should use this method if you want to restrict a user's access to any other existing or future datasets.

note

If you're granting the user access to a table, then remember that they'll be able to create views based on that dataset, which that user can then grant access to other users.

Example: Single Dataset

You have a user that you only want to give access to an individual table. You would need to navigate to the **Privileges** screen from that dataset's settings and grant the user the ``SELECT`` privilege, or perform the following command from the SQL Editor:

Single dataset example

```
GRANT SELECT ON TABLE TableA1 TO USER user1
```

The image below illustrates the objects ``user1`` now has access to.

This restricts ``user1`` so that they may only access the ``TableA1`` table, not any other datasets contained in the same folder. However, ``user1`` may still create views based on ``TableA1``.

Granting to ALL DATASETS

When you have a user that needs access to all existing datasets, then you would use the SQL syntax ``ON ALL DATASETS`` (see the example scenario outlined below). This gives the user access to all existing datasets. The user would not, however, automatically receive access to any future datasets created by other users.

You should use this method of privilege assignment if you want to restrict a user's access to parent objects, but still wish for them to have access to all existing datasets.

Example: All Datasets

You have a specific user that needs access to all datasets in a specific folder, but they do not require privileges for the folders containing these tables. You would then execute the following command from the SQL Editor:

All datasets example

```
GRANT SELECT ON ALL DATASETS IN SYSTEM TO USER user1
```

The image below illustrates the objects `user1` now has access to.

This command restricts the scope of `user1` to all datasets presently found in `source1`, such as `TableC1` and `TableD1`. Should additional datasets be created in the future, `user1` will not have access to them.

Granting to a Scope

When you want to grant a user access to a parent object, such as a folder, this will also grant the user access to any datasets contained (see the example scenario outlined below).

You should use this method of privilege management if you wanted to grant a user access to all existing and future datasets contained under a parent object.

Example: Scope

This method grants a user access to all existing and future datasets contained under a specified object. To accomplish this, you need to navigate to the **Privileges** screen from that folder's settings and grant the user the `SELECT` privilege, or execute the following command from the SQL Editor:

Scope example

```
GRANT SELECT ON FOLDER Folder3 TO USER user1
```

The image below illustrates the objects `user1` now has access to.

This grants `user1` the `SELECT` privilege on `Folder3`, which means they now have access to all existing and future datasets contained in that folder and its subfolders.

Was this page helpful?

[\[Previous](#)

[Personal Access Tokens\]\(/current/security/authentication/personal-access-tokens\)](#)[\[Next](#)

[Privileges\]\(/current/security/rbac/privileges\)](#)

Object Hierarchy

Inheritance

Scope

Current vs. Future Objects

Ownership

Managed Access Spaces

View Delegation

Privileges

Granting Privileges Using the Dremio Console

Revoking Privileges Using the Dremio Console

Granting Privileges Using SQL Commands

Source:
<https://docs.dremio.com/current/security/integrations/>

Version: current [26.x]

Integrations with Third-Party Access Control and Data Governance Platforms Enterprise

Dremio supports the following third-party integrations for managing access control and data governance:

[AWS Lake Formation](#)

[Privacera](#)

[Apache Ranger: Row-Level Filtering & Column-Masking](#)

Was this page helpful?

[Previous

Inbound Impersonation](/current/security/rbac/inbound-impersonation)[Next

AWS Lake Formation](/current/security/integrations/lake-formation/)

Source:
<https://docs.dremio.com/current/security/secrets-management/>

Version: current [26.x]

Secrets Management Enterprise

When you connect to data sources in Dremio or list secrets in Dremio configuration files, you can use a secrets management service to provide sensitive information like passwords and secret access keys instead of providing it in plaintext. When Dremio needs the value of one of these secrets to authenticate to a data source or another service, Dremio retrieves the value directly from the secrets management service using the secret reference you provide.

Dremio supports the following secrets management providers:

[AWS Secrets Manager](#)

[Azure Key Vault](#)

[HashiCorp Vault](#)

Was this page helpful?

[\[Previous](#)

[Apache Ranger\]\(/current/security/integrations/row-column-policies-ranger\)\[Next](#)

[AWS Secrets Manager\]\(/current/security/secrets-management/aws-secrets-manager\)](#)

Source:
<https://docs.dremio.com/current/security/auditing/>

Version: current [26.x]

On this page

Audit Logging Enterprise

For organizations subject to compliance and regulation where auditing is regularly required, Dremio offers full audit logging, wherein all user activities performed within Dremio are tracked and traceable via the `audit.json` file. Each time a user performs an action within Dremio, such as logging in or creating a view, the audit log captures the user's ID and username, objects affected, action performed, event type, SQL statements used, and more.

Audit logging is enabled by default and is available only to users with administrative rights at the System level.

Audit Log Location

The log-file location may be configured via the `dremio.log.path` property in the [`dremio-env` file](#). You can specify their location, size, and rotation schedule.

Tracked Events and Actions

Dremio supports audit logging for the following objects (event types) and actions:

Event Type	Actions
---	---
AI_AGENT	REQUEST, RESPONSE
AUTHENTICATION	LOGIN
ENGINE	CREATE_STARTED, UPDATE_STARTED, DELETE_STARTED
ENGINE_CONFIG	CREATE, UPDATE, DELETE, START, STOP, SCALE_UP, SCALE_DOWN
ENGINE_SCALING	SCALE_UP_STARTED, SCALE_DOWN_STARTED
FOLDER	CREATE, UPDATE, DELETE
LABEL	CREATE, UPDATE, DELETE
MODEL_PROVIDER_CONFIG	CREATE, UPDATE, DELETE, SET_DEFAULT
PERSONAL_ACCESS_TOKEN	CREATE, DELETE
PHYSICAL_DATASET	CREATE, UPDATE, DELETE
PRIVILEGE	UPDATE, DELETE
QUEUE	CREATE, UPDATE, DELETE
REFLECTION	CREATE, UPDATE, DELETE
ROLE	CREATE, UPDATE, DELETE
SOURCE	CREATE, UPDATE, DELETE
SPACE	CREATE, UPDATE, DELETE
SUPPORT_SETTING	RESET, SET
UDF	CREATE, UPDATE, DELETE
USER_ACCOUNT	CREATE, UPDATE, DELETE
VIRTUAL_DATASET	CREATE, RENAME, UPDATE, DELETE
WIKI	CREATE, EDIT, DELETE

Audit Log Format

Audit logs include the following information in JSON format:

Key	Value
---	---
`timestamp`	The date and time when the event was recorded.
`userId`	The ID value associated with the user's account. The following values are placeholders that represent internal system users, which Dremio uses to log events before the user authenticates: `1` and `678cc92c-01ed-4db3-9a28-d1f871042d9f`.
`userName`	The username associated with the user account (which is typically used to log in).
`status`	The status of the action, which is typically used to indicate whether the event was approved or allowed.
`eventType`	The object or scope of the interaction that occurred.
`action`	The actual activity performed within the specified scope. This varies based on the `eventType`, but most often would be `CREATE`, `DELETE`, and `UPDATE`.
`details`	The data altered or created. This varies based on the `eventType`.

Audit Log Examples

The following examples show the types of audit records that Dremio captures and the information included in an audit entry for each event type.

AUTHENTICATION

-

ENGINE

REFLECTION

VIRTUAL_DATASET

User `dremio` logged in on the Dremio application.

The audit log would have the following information:

Authentication log

```
{
  "timestamp": "2021-11-23 16:30:53,400",
  "userContext": {
    "userId": "1",
    "userName": "$dremio$"
  },
  "status": "OK",
  "eventType": "AUTHENTICATION",
  "action": "LOGIN",
  "details": {
    "userName": "dremio",
    "userId": "",
    "source": "FLIGHT"
  }
}
```

User `dremio` created an engine called `preview`.

The audit log would have the following information:

Engine log

```
{
  "timestamp": "2025-04-07 13:25:41,193",
  "userContext": {
    "userId": "b8c3f553-93ca-4b6b-95dc-4d6c03cdb58f",
    "userName": "dremio"
  },
  "status": "OK",
  "eventType": "ENGINE",
  "action": "CREATE_STARTED",
  "details": {
    "engineId": "541bf413-b66d-4fc0-8e33-e103efdf6bdc",
    "engineName": "preview",
    "engineSize": "2XSmall",
    "resourceAllocationOffset": "reserve-2-8",
    "targetCpuCapacity": "16C",
    "autoStopDelaySecs": 3600,
  }
}
```

User `dremio` created a Reflection called `Raw Reflection (1)` in the Dremio console.

The audit log would have the following information:

Reflection log

```
{
  "timestamp": "2021-11-22 10:06:38,432",
  "userContext": {
    "userId": "6ab04602-410b-4031-87ae-2d3d5f7dc",
    "userName": "dremio"
  },
  "status": "OK",
  "eventType": "REFLECTION",
  "action": "CREATE",
  "details": {
    "reflectionId": "a5251b05-4873-4a9d-a008-303eeeeed",
    "name": "Raw Reflection (1)",
    "dataset": "7e3d4a8a-b92d-41ab-96dc-6a76a6248",
    "type": "RAW",
    "sortColumns": [
      {
        "name": "fare_amount"
      }
    ],
    "partitionColumns": [
      {
        "name": "passenger_count"
      }
    ],
    "distributionColumns": [],
    "dimensions": [],
    "measures": [],
    "displayColumns": [
      {
        "name": "passenger_count"
      },
      {
        "name": "pickup_datetime"
      },
      {
        "name": "trip_distance_mi"
      },
      {
        "name": "fare_amount"
      },
      {
        "name": "tip_amount"
      },
      {
        "name": "total_amount"
      }
    ],
    "partitiondistributionstrategy": "CONSOLIDATED",
    "arrowCachingEnabled": false,
    "targetDataset": ""
  }
}
```



```
}
```

User `dremio` issued a SQL command to `CREATE` a view in Dremio's SQL editor.

The audit log would have the following information:

View (virtual dataset) log

```
{
  "timestamp": "2021-11-17 14:31:43,594",
  "userContext": {
    "userId": "4a3ea2fa-a3f6-4adb-8852-041a28cac",
    "userName": "dremio"
  },
  "status": "OK",
  "eventType": "VIRTUAL_DATASET",
  "action": "CREATE",
  "details": {
    "id": "da08848d-d80d-4414-aaf4-40ce866ea",
    "name": "a_employees",
    "tag": "gwN/p7E5E1Q=",
    "path": "myView.a_employees",
    "sql": "SELECT      \"firstname\",      \"zipcode\"\\nFROM
\\\"local-mysql\\\".\\\"dremio_db1\\\".\\\"Employees\\\"\\nWHERE \"zipcode\" LIKE 'a%'",
    "sqlContext": "myView",
    "fields": [
      {
        "name": "firstname",
        "dataType": "TEXT"
      },
      {
        "name": "zipcode",
        "dataType": "TEXT"
      }
    ],
    "oldName": "",
    "oldPath": ""
  }
}
```

Was this page helpful?

[\[Previous](#)

[HashiCorp Vault\]\(/current/security/secrets-management/hashicorp-vault\)\[Next](#)

[Compliance\]\(/current/security/compliance\)](#)

[Audit Log Location](#)

[Tracked Events and Actions](#)

[Audit Log Format](#)

Source:
<https://docs.dremio.com/current/security/compliance>

Version: current [26.x]

On this page

Regulatory Compliance

Dremio meets the IT control requirements for several compliance frameworks and certifications, as described below.

As the [Dremio Shared Responsibility Models](#) outline, compliance is a shared responsibility between Dremio and you. The Shared Responsibility Models lay out Dremio's responsibilities for providing standards and compliance and your responsibilities for adhering to those standards.

SOC 2 Type II Report

Dremio maintains compliance with the American Institute of Certified Public Accountants (AICPA) System and Organization Controls - Trust Services Criteria, commonly known as SOC 2.

Key Benefits

[SOC 2 Type II reports](#) provide an in-depth analysis of cloud service providers regarding the safeguards a company uses to protect customer data and how these controls are performing overall. These reports are issued by independent, third-party auditors and cover the key points of Security, Availability, Confidentiality, and Privacy.

This independent assessment of Dremio Software provides a detailed report regarding the environments used to provide security and privacy of customer data overall. The report provide descriptions of these controls, the tests performed to assess their effectiveness, the results of said tests, and then an overall opinion regarding the design and operational effectiveness of the environments.

ISO 27001 Certification

ISO 27001 is an internationally recognized specification for an Information Security Management System (ISMS). ISO 27001 is the only auditable standard that deals with the overall management of information security, rather than just which technical controls to implement.

Key Benefits

Obtaining [ISO 27001:2022 certification](#) demonstrates that Dremio employs a

comprehensive framework of legal, physical, and technical controls for information risk management.

GDPR Compliance

Dremio is compliant with the storage and security of its data according to Article 27 of the General Data Protection Regulation (GDPR). Please see [Dremio's Privacy Policy](#) for additional information regarding our appointed European Data Protection Office (EDPO) in the EU.

Key Benefits

As part of the European Union, specific regulations exist that require companies to [maintain compliance with GDPR](#). This governs the way user data is stored, processed, and utilized on Dremio Software. Specifically, this prevents the exploitation of user data and standardizes the data protection laws that services must follow throughout Europe.

CCPA Compliance

Dremio maintains compliance with the California Consumer Privacy Act (CCPA), which regulates the handling of personal data and prevents any unauthorized use or sale. Please see [Dremio's Privacy Notice For California Residents](#) for additional information.

Key Benefits

Adherence to [CCPA](#) by an organization ensures that California residents have the right to opt out of having their data sold to third parties, request disclosure of data collected, and request deletion of said data.

HIPAA Compliance

Dremio is compliant with the Health Insurance Portability and Accountability Act (HIPAA), a series of federal regulatory standards that outline the lawful use and disclosure of protected health information in the United States. HIPAA compliance is regulated by the Department of Health and Human Services (HHS) and enforced by the Office for Civil Rights (OCR).

Key Benefits

Adherence to [HIPAA](#) ensures that healthcare providers, health plans, healthcare clearinghouses, and business associates of HIPAA-covered entities must implement multiple safeguards to protect sensitive personal and health information.

Was this page helpful?

[\[Previous](#)

[Audit Logging\]\(/current/security/auditing/\)](#)[\[Next](#)

[Administration\]\(/current/admin/\)](#)

SOC 2 Type II Report

Key Benefits

ISO 27001 Certification

Key Benefits

GDPR Compliance

Key Benefits

CCPA Compliance

Key Benefits

HIPAA Compliance

Key Benefits

Source:
<https://docs.dremio.com/current/security/rbac>

Version: current [26.x]

On this page

Access Control Enterprise

Dremio allows for the implementation of granular-level privileges, which defines a user/role's access privilege and available actions for specific objects, such as folders and datasets. This is called access management, and gives administrators the ability to restrict access to any object in Dremio.

Privileges – Privileges enable users to perform explicit operations on objects in Dremio. Additionally, privileges may be set on individual datasets (tables or views) or whole schemas, allowing for a simplified configuration with larger catalogs.

Row-access and column-masking policies – Row-access and column-masking policies enable you to grant users access to particular rows or columns.

The following APIs are available for managing access control:

Grants APIs

Privileges API

Roles API

Users API

A wide range of SQL commands are also available:

Privileges SQL Commands

Roles SQL Commands

Object Hierarchy

Each object resides within a container in a hierarchy of containers. The upper-most container exists as the system user, or administrator account. All other objects are contained within sources or spaces, organized into folders. The hierarchy of these objects is illustrated below.

Inheritance

The objects to which privileges are granted depend on the inheritance model. In other words, granting access to a parent object, such as a folder, also gives that user access to any existing and future datasets contained in that folder. For example, giving a user privileges to ALL DATASETS will only grant the user access to existing datasets, not the folders that contain the datasets. In comparison, granting privileges at the source level will extend that user's access to the source's existing and future folders/schema and datasets. The object to which a user's privileges are applied is also known as the scope, and follow a parent-child relationship.

By the rules of inheritance, user or group access may be granted as high or low in the object hierarchy as you wish for access to reach.

Permissions granted to an individual table or view mean that a user's access only extends to that dataset, not to the parent folder or other datasets created in the same folder. So if a user only needs access to a single dataset, administrators need only grant privileges to that object.

Consider the image above, which shows an example of object structure in Dremio. If a user is granted privileges to a single dataset, such as `TableA1`, then that is the one object they have access to. However, if a user is granted privileges at the folder level, such as `Folder1`, then that user's access extends to any existing and future child objects created, including `FolderA`, `TableA1`, `TableB1`, and so on.

note

If a user has privileges for a single table, they may create views based on that dataset, but with the user now having `ALTER` and `MANAGE GRANTS` privileges for any view. However, the user still retains the same privileges as before with the original dataset. For more information, read View Delegation.

Scope

Scope is a concept used to describe what objects a user or group has access to. Privileges are assigned by object, which ultimately determines what a grantee may perform set functions upon. For example, you may set a user's scope to `FolderA`, which will give the user access to all existing and future datasets contained in the folder, as well as the datasets' wikis. But they will not have access to any other folders or the source. The object a user is granted access to is dependent on the inheritance

model, which means based on the object type, it may contain child objects. For example, if a user is granted privilege to a folder, the user's access also extends to all existing and future datasets contained in that folder.

For example, `user1` is granted the `SELECT` privilege to the folder `FolderC`. This object contains multiple datasets, which the user may now access. However, there exists a parent folder and another subfolder with its own datasets.

Because of the established scope, `user1` may not access `FolderD` because they were only granted access to `FolderC`'s objects.

Current vs. Future Objects

Based on the selected scope, you may restrict a user's access to future and existing datasets. For example, if you select a single table as the scope of a user's privilege, then that user may only perform that action to the existing dataset, as well as any future views they create using that table. However, they may not access any views created from a table by another user (see the example below). However, if the scope is instead set at the folder level, then the user may perform the granted privilege to all tables and views contained in that folder (see the example below).

Ownership

Object ownership is a security feature used to control access to an object. In Dremio, each object must have an owner, and may have only one owner. Ownership is automatically granted to the user who initially created the object. For example, when `user1` creates an S3 data source, Dremio automatically assigns ownership of the source to `user1`.

The privileges included in object ownership depend on your configuration.

By default, ownership includes all privileges for that object. The object owner can grant or revoke access privileges to the object and its child objects, modify an object's settings, and delete the object as desired. See [Granting Privileges Using SQL Commands](#) for more information.

Managed access spaces centralize the administration of access privileges in shared spaces to a limited set of users and roles, including the space owner. By limiting privilege grant authority, managed access spaces help ensure consistent and controlled access policies and reduce the risk of unauthorized access. See [Managed Access Spaces](#).

The following behaviors and limitations apply to ownership:

Each object may only have one owner.

An object's creator is automatically granted ownership.

Object ownership may be assigned or modified to a new user or role with the `GRANT OWNERSHIP` command.

The object's access control settings may not work if the owner is deleted or removed. See [View Delegation](#).

Object owners may be identified by querying the ``sys."tables"`` table or ``sys.views`` table. If an object has no owner, the ``owner_id`` will display as ``$unowned``.


Managed Access Spaces

Managed access spaces centralize the administration of access privileges to a limited set of users and roles, including:

Owner of the space

Dremio administrator

Users or roles explicitly granted the `MANAGE GRANTS` privilege on the object or any of its parents

When using managed access spaces, Dremio displays shared spaces with a lock icon  on the [Datasets](#) and [SQL Runner](#) pages. Owners of folders, views, and functions in a locked space cannot grant or revoke privileges on those objects to other users or roles.

User or role	Grants/revokes privileges in default spaces	Grants/revokes privileges in managed access spaces
---	---	---
Dremio administrator	Yes	Yes
Owner of a shared space	Yes	Yes
Owner of a folder in a shared space	Yes	No
Owner of a view in a shared space	Yes	No
Owner of a function in a shared space	Yes	No
User or role with <code>MANAGE GRANTS</code>	Yes	Yes

Managed access spaces do not impact:

Any user home space

Sources, including Nessie catalogs

Global objects, such as scripts and user-defined functions

Managed access spaces do not override a `MANAGE GRANT` privilege granted at system scope.

The Dremio administrator can activate managed access spaces by setting the ``security.access-control.managed-access-spaces.enabled`` [support key](#) on the Support Settings page.

View Delegation

View delegation means that the data in tables with restricted access may be available to other Dremio users by creating views. View delegation is the critical capability of the Dremio semantic layer that allows users to run queries without accessing the underlying tables and views directly.

The fundamental principles of view delegation include the following:

The privileges of a view's owner determine whether the view can use dependent tables

and views.

Additional user access to a view is controlled by privilege grants directly on the view, forming a privilege chain from the view to the underlying table.

note

A shared view selects from the underlying dataset using the view owner's permissions at the time of the view's last modification, even if the end user querying the view lacks privileges to modify the underlying table. This applies to each table on the data graph and chain of datasets.

View delegation is different from privilege assignment. View delegation is an implicit delegation of the SELECT privilege on underlying objects, which means that users who run queries on a view must have access privileges on the view but do not need privileges on underlying tables. Privilege assignment is an explicit delegation providing direct access to an object.

Example 1: View Delegation

`user1` has the SELECT privilege on `table1` and creates `view1` to filter and transform data in `table1`. `user2` asks for access privileges to run queries on `view1` as well. `user2` may obtain the SELECT privilege for `view1` from the following authorized users:

By default, view owners such as `user1` can grant and revoke privileges to other users, as appropriate.

A limited set of users and roles, such as the space owner, can grant or revoke privileges in managed access spaces.

Dremio administrators or other users with the MANAGE GRANTS privilege can grant privileges to other users.

If access for `user2` is appropriate, the authorized user runs `GRANT SELECT ON VIEW view1 TO USER user2` to grant the SELECT privilege to `user2`. After `user2` obtains the SELECT privilege, they can run queries on `view1`, utilizing the privilege of `user1` as owner to `view1` to SELECT from `table1`.

Object	`user1`		`user2`	
---	---		---	
`view1`	OWNERSHIP		SELECT	
`table1`	SELECT		None	

Privileges by user in Example 1

The following table describes the actions that each user may perform based on their privileges:

Task	Works for `user1`		Works for `user2`	
---	---		---	
Use `view1` in queries	Yes, `user1` owns `view1` and has the SELECT privilege on `table1`. Yes, `user2` has the SELECT privilege on `view1` and the owner of `view1` has the SELECT privilege on `table1`.			
Modify the query in `view1`	Yes, `user1` is the owner of `view1`. Ownership includes the ALTER privilege to modify the view definition. No, `user2` is not the owner of			

`view1` and does not have the ALTER privilege. |
| Use `table1` in queries | Yes, `user1` has the SELECT privilege on `table1`. | No,
`user2` has no privileges on `table1` and cannot see it in the Dremio user interface. |

Tasks by user in Example 1

Example 2: View Delegation with Revoked Access to the Original Table

To continue the previous example, `user1` has SELECT access to `table1`, which gives `user1` access through `view1`. An administrator revokes the SELECT access of `user1` on `table1`.

Object	`user1`	`user2`
---	---	---
`view1`	OWNERSHIP	SELECT
`table1`	None	None

Privileges by user in Example 2

The following table describes the actions that each user may perform based on their privileges:

Task	Works for `user1`	Works for `user2`
---	---	---
Use `view1` in queries	No, `user1` no longer has SELECT on the underlying `table1`.	No, `user2` no longer has a chain of permission through `user1` to `table1`.
Modify the query in `view1`	No, `user1` is the owner of `view1` but any attempts to edit `view1` will fail unless the references to `table1` are removed since `user1` can no longer access `table1`.	No, `user2` is not the owner of `view1` and does not possess the ALTER privilege.
Use `table1` in queries	No, `user1` has no privileges on `table1`.	No, `user2` has no privileges on `table1`.

Tasks by user in Example 2

Privileges

Privileges refer to the defined levels of access or permissions that are assigned to roles or users within Dremio. Privileges determine the operations a user or role can perform on securable objects. Examples of privileges in Dremio include SELECT on a table or view, INSERT on a table, DELETE on a table, CREATE TABLE on a folder, and MANAGE GRANTS on any object.

The assignment of privileges to roles, or users, should be based on the principle of least privilege, where users or roles are given only the minimum privileges required to perform their tasks effectively.

Privileges can be managed using SQL, APIs, or the Dremio Console.

For more information, please refer to [Privileges](#).

Granting Privileges Using the Dremio Console

You can share catalog objects with others in your organization by granting privileges on the objects to users and roles as follows:

Locate the desired object.

Click !This is the icon that represents more actions. or !This is the icon that represents more actions. depending on the object.

In the object settings dialog, select **Privileges** from the settings sidebar.

note

For some object types, the settings dialog automatically opens to display the privilege settings, and you do not need to select the Privileges tab.

In the Privileges dialog, in the field under **Add User/Role**, enter the exact names of the users and roles to which you want to grant privileges.

note

Because all users are members of the PUBLIC role, you can use the PUBLIC role to grant privileges to all users.

Click **Add to Privileges**.

For each entry in the **Add User/Role** field that matches a user or role in Dremio, a record appears in the USERS/ROLES table.

In the USERS/ROLES table, toggle the checkbox for each privilege you want to grant for that user or role. For a description of the privilege, hover over the column name in the USERS/ROLES table. See the example below:

![/assets/images/privileges-users-table-5eb046e1d2acd0cc1a6d1bd87d3afc35.png)

(Optional) Repeat steps 4-6 if you want to add more users or roles and grant them privileges.

When finished, click **Save**.

Revoking Privileges Using the Dremio Console

To revoke user and role privileges, complete the following steps:

Locate the desired object.

Click !This is the icon that represents more actions. or !This is the icon that represents more actions. depending on the object.

In the object settings dialog, select **Privileges** from the settings sidebar.

note

For some object types, the settings dialog automatically opens to display the privilege settings, and you do not need to select the Privileges tab.

In the USERS/ROLES table, locate the desired user or role record. If the user or role is not listed, then they do not have specific privileges on the object.

To revoke some but not all privileges for the user or role, clear the checkboxes in the columns for the privileges you wish to revoke.

To revoke all privileges for a user or role, click !This is the icon that represents more actions. next to the user or role name and select **Remove**.

For a description of the privilege, hover over the column name in the USERS/ROLES table. See the example below:

When finished, click **Save**.

note

If a user has a specific privilege on an object through their memberships in multiple roles and the privilege is revoked for one of the roles, the user retains the privilege until it is revoked on the same object for all roles to which the user belongs.

tip

You can also grant or revoke privileges using SQL commands or APIs.

Granting Privileges Using SQL Commands

When granting privileges to users and roles with SQL commands, you may follow one of three methods: granting to a single dataset, granting to ALL DATASETS, and granting to a scope. Examples of these methods may be found under each section.

Each example includes an SQL command. For more information about command syntax, review the Privileges (GRANT/REVOKE) SQL commands.

note

Because all users are members of the PUBLIC role, you can use the PUBLIC role to grant privileges to all users.

Granting to a Single Dataset

When you have a user that needs access to only one table and no other objects, then you would simply assign them privileges for that dataset (see the example scenario outlined below).

You should use this method if you want to restrict a user's access to any other existing or future datasets.

note

If you're granting the user access to a table, then remember that they'll be able to create views based on that dataset, which that user can then grant access to other users.

Example: Single Dataset

You have a user that you only want to give access to an individual table. You would need to navigate to the *Privileges* screen from that dataset's settings and grant the user the `SELECT` privilege, or perform the following command from the SQL Editor:

Single dataset example

```
GRANT SELECT ON TABLE TableA1 TO USER user1
```

The image below illustrates the objects `user1` now has access to.

This restricts `user1` so that they may only access the `TableA1` table, not any other datasets contained in the same folder. However, `user1` may still create views based on `TableA1`.

Granting to ALL DATASETS

When you have a user that needs access to all existing datasets, then you would use the SQL syntax `ON ALL DATASETS` (see the example scenario outlined below). This gives the user access to all existing datasets. The user would not, however, automatically receive access to any future datasets created by other users.

You should use this method of privilege assignment if you want to restrict a user's access to parent objects, but still wish for them to have access to all existing datasets.

Example: All Datasets

You have a specific user that needs access to all datasets in a specific folder, but they do not require privileges for the folders containing these tables. You would then execute the following command from the SQL Editor:

All datasets example

```
GRANT SELECT ON ALL DATASETS IN SYSTEM TO USER user1
```

The image below illustrates the objects `user1` now has access to.

This command restricts the scope of `user1` to all datasets presently found in `source1`, such as `TableC1` and `TableD1`. Should additional datasets be created in the future, `user1` will not have access to them.

Granting to a Scope

When you want to grant a user access to a parent object, such as a folder, this will also grant the user access to any datasets contained (see the example scenario outlined below).

You should use this method of privilege management if you wanted to grant a user access to all existing and future datasets contained under a parent object.

Example: Scope

This method grants a user access to all existing and future datasets contained under a specified object. To accomplish this, you need to navigate to the **Privileges** screen from that folder's settings and grant the user the ``SELECT`` privilege, or execute the following command from the SQL Editor:

Scope example

```
GRANT SELECT ON FOLDER Folder3 TO USER user1
```

The image below illustrates the objects ``user1`` now has access to.

This grants ``user1`` the ``SELECT`` privilege on ``Folder3``, which means they now have access to all existing and future datasets contained in that folder and its subfolders.

Was this page helpful?

[\[Previous](#)

[Personal Access Tokens\]\(/current/security/authentication/personal-access-tokens\)](#)

[\[Next Privileges\]\(/current/security/rbac/privileges\)](#)

[Object Hierarchy](#)

[Inheritance](#)

[Scope](#)

[Current vs. Future Objects](#)

[Ownership](#)

[Managed Access Spaces](#)

[View Delegation](#)

[Privileges](#)

[Granting Privileges Using the Dremio Console](#)

[Revoking Privileges Using the Dremio Console](#)

[Granting Privileges Using SQL Commands](#)

Source:
<https://docs.dremio.com/current/security/integrations/lake-formation/>

Version: current [26.x]

On this page

Integrate with AWS Lake Formation Enterprise

Lake Formation provides access controls for datasets in the AWS Glue Data Catalog and is used to define security policies from a centralized location that may be shared across multiple tools. Dremio may be configured to refer to this service to verify access for a user to contained datasets.

Requirements

Dremio v19.0+

Identity Provider service (e.g., Microsoft Entra ID, LDAP) set up

(Recommended) SAML connection with AWS

Permissions set up in Lake Formation

AWS Glue Data Catalog connected to Dremio

User and Group ARN prefixes specified and enabled

Lake Formation Workflow

When Lake Formation is properly configured, Dremio adheres to the following workflow each time an end user attempts to access, edit, or query datasets with managed privileges:

Dremio enforces access control. See Configuring Sources for Lake Formation below for access control recommendations.

Dremio checks each table to determine if those stored in the AWS Glue source are configured to use Lake Formation for security.

If one or more datasets leverage Lake Formation, Dremio determines the user ARNs to use when checking against Lake Formation.

Dremio queries Lake Formation to determine a user's access level to the datasets using the user/group ARNs.

If the user has access to the datasets specified within the query's scope, the query proceeds.

If the user lacks access, the query fails with a permission error.

Demoing Lake Formation

Demo files and a walkthrough are available to help you test Lake Formation functionality. The demo files and walkthrough are intended for users who have not configured all of the requirements listed above.

Configuring Sources for Lake Formation

Lake Formation integration is dependent on the mapping of user/group names in

Requirements

Lake Formation Workflow

Demoing Lake Formation

Configuring Sources for Lake Formation

Lake Formation Cell-Level Security

Source:
<https://docs.dremio.com/current/security/integrations/privacera/>

Version: current [26.x]

On this page

Integrate with Privacera Enterprise

Dremio and Privacera have partnered to provide an integration that allows organizations to implement fine-grained access controls on their open data lakehouse. The integration provides the following capabilities:

Privacera Policy Sync: Data governance and access control policies in Privacera are translated into SQL and pushed to Dremio using Dremio's native RBAC and fine-grained access control, ensuring that data remains secure and compliant with the centrally-defined policies in Privacera. You can learn more about how this works in Privacera's [Dremio connector](#) documentation.

Dremio Auditing and Query Tracking: Audit details related to user-executed queries in Dremio can be accessed through the Jobs page in the Dremio console. Job lists and job details provide insights into user-executed queries. See [Viewing Jobs](#) for more information.

Support for all Data Sources: The Privacera integration supports all Dremio data sources. See [Connect to Your Data](#) for a full list of sources.

note

When using the Privacera plugin for Dremio, no additional or external tools should be used for the policy synchronization between Privacera and Dremio.

Prerequisites

Ensure that you meet the following prerequisites before you begin the integration:

An on-premise or SaaS Privacera Manager host that is running Privacera services

A deployment of Dremio Enterprise Edition 24.1 or later Community Edition and Dremio Cloud *are not* supported at this time

Installation

Refer to the [Privacera documentation](#) to learn how to install and configure Privacera's plugin for Dremio.

note

After installing the Privacera plugin, ensure that the **Enable external authorization plugin** option is selected under **Settings > Advanced Options** on all sources that should utilize the integration with Privacera. After updating any source configurations, restart Dremio.

Was this page helpful?

[Previous

Lake Formation Demo](/current/security/integrations/lake-formation/lake-formation-demo)[Next

Apache Ranger](/current/security/integrations/row-column-policies-ranger)

Prerequisites

Installation

Source:
<https://docs.dremio.com/current/security/integrations/row-column-policies-ranger/>

Version: current [26.x]

On this page

Apache Ranger: Row-Level Filtering & Column-Masking Enterprise

Dremio offers both Apache Ranger security policy support and built-in SQL functions for applying row-level filtering and column-masking.

Column-Masking Overview

Column-masking is a secure and flexible resource-based solution to hiding sensitive information rapidly on a Hive source. Via Apache-Ranger-based security policies or using Dremio's built-in masking, you may mask or scramble private data at the column-level in a dynamic fashion for Hive query outputs. Utilizing masking methods, you may set a column to only display the year of a data, the first or last four digits of a value, and more.

Utilizing services like Apache Ranger allow you to apply access policies to a Hive source so that filters may be based upon specific users, groups, and conditions. Thus, sensitive information never leaves the source and no changes are required by the source. This likewise removes the need to produce a secondary set of data with protected information manually removed.

The following conditions apply to column-masking:

Multiple masking types are available

Masks may be applied to users, groups, and conditions

Each column must have its own masking policy

Masks are evaluated in the order they are presented in a query or on a security policy

Wildcard matching is not supported

For Apache Ranger implementations, additional use cases may be found at [3. Use cases: data-masking](#).

Row-Level Filtering Overview

Row-level filtering both simplifies queries and adds a layer of security to the data returned for user/role queries. Either SQL functions or Apache-Ranger-based security policies limit access down to the dataset layer, which then affects how queries are handled upon execution. Row-level security on supported tables helps reduce exposure of sensitive data to specific users or groups. Row segmentation and restricted access together ensures that upon query completion, only specific rows based on both the user's characteristics (username or group/role) and the runtime context of the query are displayed from Dremio's SQL Editor.

Row-level restrictions may be set by user, group/role, and other conditions (conditions only available for Ranger implementations, as described further under Row Filter Conditions).

The following examples serve as use cases where row-level filtering would prove beneficial:

Hospitals may create security policies enabling 1) doctors to view only the rows containing their patients, 2) insurance claims adjusters to view only rows pertaining to their site/facility, and 3) medical billing coders to only view rows pertaining to specific medical disciplines.

Financial institutes may create policies restricting access to rows pertaining to a user's specific division, geographic location or site, or role, meaning only employees in Collections would only be allowed to see outstanding unpaid claims, collection payment plans, and so on.

Organizations utilizing multi-tenant applications may use row-level filters to set logical separations of each tenant's data, thus ensuring a tenant only has access to their own data rows.

For Apache Ranger implementations, additional use cases may be found at [2. Use cases: row-level filters](#).

Using Apache Ranger Security Policies

For organizations configured to use [Apache Ranger](#) and Hive sources, support automatically exists in Dremio to handle security policies set from Ranger. Based on the user, group/role, and conditions set externally, Dremio automatically applies restrictions to a user's query and applies row-level filtering and column-masking in the

background. Upon query completion, you will then only see the results for rows and columns you have access to, without any visual indication that rows have been removed from view.

Requirements

Dremio 20.0+

Apache Ranger configured

Admin privileges to add access control policies

Hive source

How It Works

Ranger-based row filtering and column-masking functions as an "implicit view," replacing a table/view reference in an SQL statement prior to processing the statement. This implicit view is created through an examination of user permissions. For example, consider a user with access to `table_1`, while also having a mask applied on `table_1.column_1`, effectively translating the column to "xxx." Simultaneously, a row filter exists for `table_1.column_2`.

The original query would appear as:

Original query

```
SELECT column_1
FROM table_1
WHERE column_3
```

With both column-masking and row-level filtering policies applied from Ranger, the query above is rewritten to the following:

Query with column-masking and row-level filtering policies

```
WITH filtered_and_masked_table_1
AS (
  SELECT 'xxx' AS column_1, column_2, column_3
  FROM table_1
  WHERE column_2
)
SELECT column_1
FROM filtered_and_masked_table_1
WHERE column_3;
```

Setting Policies in Apache Ranger

For organizations currently utilizing Apache Ranger and configured to apply policies to Dremio, the application of row-level filtering and column-masking is automatic. However, in order to apply these security measures, you must also create security

policies from Ranger, which will then propagate down to Dremio when the affected users perform a query.

To create a security policy in Apache Ranger:

Navigate to the **Service Manager** page, and then select the desired **Hive Service**.

Click the **Column Masking** or **Row Level Filter** tab.

Click **Add New Policy**.

Now you are at the **Add Policy** screen. The sections below describe the elements contained on that page.

Policy Details

The following table describes the **Policy Details** section of the **Create Policy** screen.

Field	Required	Description
Policy Name	YES	The name of the policy. This value cannot be duplicated in another policy.
Policy Label		Tags to help categorize and make the policy more searchable.
Hive Database	YES	The name of the database(s) to which this policy applies. The field will display auto-complete options based on what matches the current entered value. The database must be a parent to any specified table(s) below, otherwise it will fail to apply.
Hive Table	YES	The name of the table(s) to apply the policy toward. Please ensure the tables are associated with the database(s) specified above, otherwise they will not be accessible.
Description		A description of the policy to explain its intended purpose, its audience, and any other relevant details.
enabled/disabled	YES	Determines whether the specific policy apply to the specified users, groups/roles, and conditions. If disabled, the security policy will not affect user queries.
normal/override	YES	Controls how the policy is prioritized against other existing security policies. If set to override , this policy will ignore other policies that may restrict or grant access beyond the scope specified here.
Audit Logging	YES	Controls whether auditing is enabled and is set to YES by default. Auditing tracks all user actions impacted by this policy.

Row Filter Conditions

The following table describes the **Row Filter Conditions** section of the **Create Policy** screen.

Field	Description
Select Group	The group(s) of users to which this policy applies. The public group will apply to all users. If no group is specified, a user must be provided.

| Select User | The individual user(s) to which this policy applies. If no user is specified, a group must be provided. |

| Access Types | The action which the specified group(s) or user(s) may utilize from the Dremio SQL Editor. Currently, the only type available is select. This is used in tandem with the WHERE clause as specified in the Row Level Filter field. |

| Row Level Filter | A valid WHERE clause as entered in the Enter filter expression pop-up upon clicking the Add Row Filter button. To allow full SELECT access to users without row-level filtering, do not click this button. Filters are applied based on top-down order, meaning the filter at the top is applied first, then the second filter, and so on. |

Mask Conditions

| Field | Description |

| --- | --- |

| Select Group | The group(s) of users to which this policy applies. The public group will apply to all users. If no group is specified, a user must be provided. |

| Select User | The individual user(s) to which this policy applies. If no user is specified, a group must be provided. |

| Access Types | The action which the specified group(s) or user(s) may utilize from the Dremio SQL Editor. Currently, the only type available is select. This is used in tandem with the WHERE clause as specified in the Row Level Filter field. |

| Select Masking Type | The type of column-masking behavior to apply to the associated users/groups when they query the table specified on this policy. * **Redact** - Replaces all alphabetic characters with `x` and all numeric characters with `n`. * **Partial mask: show last 4** - Displays only the last four characters of the full column value's. * **Partial mask: show first 4** - Displays only the first four characters of the full column value's. * **Hash** - Replaces all characters with a hash of the entire cell's value. * **Nullify** - Replaces all characters in the cell with a `NULL` value. * **Unmasked (retain original value)** - No masking is applied to the cell. * **Date: show only year** - Displays the year portion of a date string, defaulting the month and day to `01/01`. * **Custom** - Specifies a custom column masked value or valid Dremio expression. Custom masking may not use

[Hive UDFs]<<https://cwiki.apache.org/confluence/display/hive/languagemanual+udf#LanguageManualUDF-DataMaskingFunctions>>. Masks are applied based on top-down order, meaning the mask at the top is applied first, then the second mask, and so on. |

Adding a Row-Level Filter Policy

This section outlines how to create a row-level filter policy from the Apache Ranger console.

For additional instructions and information about row-level filtering, see [Row-level filtering and column-masking using Apache Ranger policies in Apache Hive](#).

To create a policy that enforces row-level access control, perform the following steps:

From the Apache Ranger console, navigate to the **Service Manager** page, and then select the desired **Hive Service**.

Click the **Row Level Filter** tab.

Click **Add New Policy**.

From the *Create Policy* page, provide values for the **Policy Details** and **Row Filter Conditions** sections.

Add any desired conditions, or else leave the **Row Filter Conditions** section blank to apply no filtering.

To move a condition under the **Row Filter Conditions** section, click the dotted icon on the left-hand side of the row, and then drag it to the desired new location,

Click **Add** to save the new policy.

Adding a Column-Masking Policy

This section outlines how to create a column-masking policy from the Apache Ranger console.

For additional instructions and information about column-masking, see [Row-level filtering and column-masking using Apache Ranger policies in Apache Hive](#).

To create a policy that enforces row-level access control, perform the following steps:

From the Apache Ranger console, navigate to the *Service Manager* page, and then select the desired **Hive Service**.

Click the **Row Level Filter** tab.

Click **Add New Policy**.

From the *Create Policy* page, provide values for the **Policy Details** and **Mask Conditions** sections.

Create any desired masking conditions under the **Mask Conditions** section, or else select **Unmasked (retain original value)** to not apply masking for a user or group.

To move a condition under the **Mask Conditions** section, click the dotted icon on the left-hand side of the row, and then drag it to the desired new location,

Click **Add** to save the new policy.

Using Dremio's Built-In Filtering/Masking

For organizations not using Apache Ranger, Dremio offers column-masking and row-level filtering for views via SQL functions. However, this implementation is limited in comparison to the security policies possible with [Ranger implementations](#). Where possible, utilize this service to enforce row-level permissions and column-masking as described above.

note

We recommend using Dremio 20.0+ in tandem with Apache Ranger to apply user/role-based security policies across all datasets while querying a table/view. Otherwise, you may utilize Dremio's built-in SQL functions (as describe below) to manually enforce filtering and masking.

Creating a View with Column-Masking

By using the query_user() or ``is_member()`` SQL functions, a view can be configured manually to allow selective masking of columns for different users/roles without the need to create multiple datasets.

The following is a sample SQL command for a view using column-masking syntax:

Example for view using column-masking

```
SELECT
  CASE
    WHEN query_user() IN ('dave','mike') OR is_member('Accounting') THEN SSN
    ELSE CONCAT('XXX-XX-',SUBSTR(SSN,8,4))
  END
FROM ss.crm.dbo.employees
```

The SQL function ``is_member()`` is case-insensitive by default. This may be circumvented by adding a boolean ``is_member(groupname, <case-sensitivity boolean>)`` to control case-sensitivity. Simply set it to ``true`` to enable case-sensitivity or ``false`` to disable. If omitted from the SQL command, the boolean defaults to ``false``.

Creating a View with Row-Level Permissions

By using the query_user() or ``is_member()`` SQL functions, a view can be configured to allow manual selective filtering of rows for different users/roles without the need to create multiple datasets.

The following is a sample SQL command for a view using row-level filtering syntax:

Example for view using row-level filtering

```
SELECT *
FROM mongo.view.business
WHERE
  (state = 'NV' AND query_user() IN ('dave','mike'))
OR
  (state = 'CA' AND is_member('Marketing'))
```

The SQL function ``is_member()`` is case-insensitive by default. This may be circumvented by adding a boolean ``is_member(groupname, <case-sensitivity boolean>)`` to control case-sensitivity. Simply set it to ``true`` to enable case-sensitivity or ``false`` to disable. If omitted from the SQL command, the boolean defaults to ``false``.

Was this page helpful?

[Previous

Privacera](/current/security/integrations/privacera)[Next

Secrets Management](/current/security/secrets-management/)

Column-Masking Overview

Row-Level Filtering Overview

Using Apache Ranger Security Policies

Requirements

How It Works

Setting Policies in Apache Ranger

Policy Details

Row Filter Conditions

Mask Conditions

Adding a Row-Level Filter Policy

Adding a Column-Masking Policy

Using Dremio's Built-In Filtering/Masking

Creating a View with Column-Masking

Creating a View with Row-Level Permissions

Source:
<https://docs.dremio.com/current/security/authentication/identity-providers/ldap>

Version: current [26.x]

On this page

LDAP Enterprise

Configuring Dremio for LDAP

To configure Dremio for LDAP, perform the following steps:

Create a new `ad.json` file that contains your LDAP server configuration. See the LDAP Properties below for more information.

Adding your configuration:

Kubernetes

Standalone

Update the `coordinator.web.auth.type` configuration in your `values-overrides.yaml` with the value `ldap`. See the configuration of [Identity Providers](#).

Optionally, to configure Dremio to use TLS when connecting to LDAP, perform the following steps:

Configure the LDAP `connectionMode` in `ad.json` for the required level of TLS functionality. See [LDAP Connection Mode](#).

To configure a truststore for the validation of TLS LDAP certificates, add the following to `values-override.yaml`

New configuration for TLS to LDAP

```
dremio:
  advancedConfigs:
    trustStore:
      enabled: true
      password: "changeit"
```

Add the `ad.json` file to your Dremio deployment. This can be done in one of two ways:

Method 1 (Preferred)

Add the content of your JSON file into your `values-override.yaml` via the `ssoFile` option. This method is detailed in the [Identity Provider](#) section.

If TLS with a custom truststore is required, use the `configBinaries` option in your `values-overrides.yaml` and pass in the content of your `.jks` truststore file. For more details, see [Additional Config Binary Files](#).

Method 2

Perform a `helm install` with the `set-file coordinator.web.auth.ssoFile=<your-local-path>/ad.json` option indicating the location of the `ad.json`. See [Deploying Dremio to Kubernetes](#) for additional information.

Use `set-file dremio.configBinaries.trustStore=<path/to/truststore/jks/file/on/local/machine>` to pass in a local truststore file, if TLS is required.

Edit the `dremio.conf` file, and add the following properties:

Example Dremio Service Configuration

```
services: {
  coordinator.enabled: true,
  coordinator.web.auth.type: "ldap",
  coordinator.web.auth.config: "ad.json"
}
```

Optionally, to configure Dremio to use TLS when connecting to LDAP, perform the following steps:

Configure the LDAP `connectionMode` in `ad.json` for the required level of TLS

functionality. See LDAP Connection Mode.

To configure a truststore for the validation of LDAP TLS certificates, update ``dremio.conf`` with ``javax.net.ssl`` settings for the trustStore and trustStorePassword.

Example Truststore Configuration

```
javax.net.ssl {  
    trustStore: "<path/to/truststore/jks/file>",  
    trustStorePassword: "trustStorePassword"  
}
```

Copy the modified ``dremio.conf`` and ``ad.json`` files to every coordinator node in the Dremio cluster. The location of the ``ad.json`` file is relative to the ``/conf`` directory. The path to the file can be absolute; the file can live anywhere in the system.

Scale-Out Coordinators

When using scale-out coordinators, you must ensure that both the ``dremio.conf`` configuration and the ``ad.json`` file are present on every coordinator node. Scale-out coordinators require the authentication configuration even when ``coordinator.web.enabled: false`` is set.

LDAP Properties

The ``ad.json`` file is a JSON-formatted config file that defines how Dremio connects to and communicates with your LDAP/AD server, including how it finds users, groups, and handles secure authentication.

Example Configuration for LDAP using Group List

```
{  
    "connectionMode": "PLAIN",  
    "servers": [  
        {  
            "hostname": "ldap.example.com",  
            "port": 389  
        }  
    ],  
    "names": {  
        "bindDN": "CN=admin,DC=drem,DC=io",  
        "bindMethod": "UNAUTHENTICATED",  
        "bindPassword": "admin",  
        "baseDN": "dc=drem,dc=io",  
        "userFilter": "&(objectClass=posixAccount)",  
        "userAttributes": {  
            "baseDNs": [  
                "OU=Users,OU=ldaptest,DC=drem,DC=io",  
            ],  
            "id": "uid",  
            "firstname": "givenName",  
            "lastname": "sn",  
            "email": "mail"  
        }  
    },  
}
```

```

    "userGroupRelationship": "GROUP_ENTRY_LISTS_USERS",
    "groupEntryListsUsers": {
        "userEntryUserIdAttribute": "uid",
        "groupEntryUserIdAttribute": "memberUid"
    },
    "groupDNs": ["cn={0},OU=test,OU=ldaptest,DC=drem,DC=io",
        "cn={0},OU=dev,OU=ldaptest,DC=drem,DC=io"],
    "groupFilter": "(objectClass=posixGroup)(objectClass=sub)",
    "autoAdminFirstUser": false
}
}

```

LDAP Connection Mode

The `connectionMode` property configures how Dremio establishes connections to the LDAP/Active Directory servers. The two main secure options — `ANY_SSL` and `TRUSTED_SSL` — both use SSL/TLS but differ in how SSL certificates are validated. The modes are:

`PLAIN`: The connection between Dremio and the LDAP server is unencrypted. Dremio connects over port 389 by default, the standard LDAP port for unencrypted communication. This mode is appropriate for internal networks and isolated or trusted environments.

`ANY_SSL`: Encrypts the connection using SSL/TLS. This mode does not validate the LDAP server's SSL certificate, so it is useful for testing or internal environments where strict certificate checks are not required.

`TRUSTED_SSL`: This mode encrypts the connection using SSL/TLS and validates the LDAP server's SSL certificate against the Java truststore. This mode requires additional configuration in `dremio.conf` with the location of the trust store and its password.

LDAP Server Configuration

The `servers` section of an `ad.json` file defines the LDAP servers that Dremio can use for authentication and directory lookups. Each server accepts the following properties:

`hostname`: The Fully Qualified Domain Name or IP address of the LDAP server.

`port`: The port where the LDAP server accepts connections. Port 389 is the default LDAP when the `connectionMode` is `PLAIN`; port 636 is the default port when using SSL/TLS.

LDAP User and Groups

The `names` section maps LDAP attributes to Dremio's internal user and group fields. LDAP `names` are defined using the following properties:

Property	Required	Description
<code>autoAdminFirstUser</code>	No	The first valid LDAP user to log in to Dremio is given the

Admin role by default. This behavior, defined by ``autoAdminFirstUser: true``, is included in the ``ad.json`` file. Alternatively, you can specify a list of users and/or groups to be given the Admin role during initial login; it is used for bootstrapping only. See Admin Users for additional configuration information. |

| ``baseDN`` | Yes | A base distinguished name is the search's root path. If ``userAttributes.baseDNs`` or ``groupAttributes.baseDNs`` are specified, they override ``baseDN`` for search purposes. |

| ``bindDN`` | No | A bind distinguished name is a client's username to authenticate (bind) to the LDAP directory server. This property is not required when using a ``bindMethod`` of ``ANONYMOUS``. In particular, ``CN=admin,DC=drem,DC=io`` must not be used. |

| ``bindMethod`` | No | The authentication method: * ``ANONYMOUS``: Connect anonymously to the LDAP server. When authenticating to Dremio, empty passwords for users are not allowed. * ``SIMPLE_BIND``: Default. Connect and authenticate to the LDAP server using ``bindDN`` and ``bindPassword``. * ``UNAUTHENTICATED``: Connect to the LDAP server using an unauthenticated bind. ``bindDN`` is required. |

| ``bindPassword`` | No | Password credential for the user who connects from the Dremio LDAP client to the LDAP server. ``bindPassword`` can be encrypted using the ``dremio-admin encrypt`` CLI command. This property must not be present if you are using ``ANONYMOUS`` or ``UNAUTHENTICATED`` for ``bindMethod`` mode. See Bind Password Options for additional configuration information. |

| ``email`` | No | Attribute for the email address. |

| ``firstname`` | No | Attribute for the first name. |

| ``groupAttributes`` | No | A mapping of LDAP group attributes to Dremio group attributes. The ``baseDN``, ``searchScope``, and ``id`` properties are used. |

| ``groupDNs`` | No | A group distinguished name refers to the full path of a specific group object used for organizing users. |

| ``groupFilter`` | Yes | LDAP filter for groups. |

| ``groupMembership`` | No | Value returned by the Dremio ``memberOf()`` function. This attribute specifies the groups containing a user or a group. |

| ``groupRecursive`` | No | Attribute of a user or a group that lists transitive group membership. |

| ``id`` | No | If used with the ``userAttributes`` property, ``id`` is the attribute for the login name, defaulting to ``sAMAccountName``. If used with the ``groupAttributes`` property, ``id`` is the attribute for the group name, defaulting to ``CN``. |

| ``lastname`` | No | Attribute for the last name. |

| ``searchScope`` | No | Scope of user searches: * ``BASE``: Match the exact entry. * ``ONE``: Searches immediate children below the specified ``baseDN``. * ``SUB_TREE``: Default. Searches subtrees below the specified ``baseDN``. |

| ``userAttributes`` | No | A mapping of LDAP user attributes to Dremio user attributes. This property should include ``firstname``, ``lastname``, and ``email``. |

| ``userDNs`` | No | A user distinguished name is the unique path that identifies a specific user object. |

| ``userFilter`` | Yes | LDAP filter for validating users. Only users who fit the specific criteria are allowed to authenticate. |

| ``userGroupRelationship`` | No | Determines whether you are implementing lists based on users or groups. * ``GROUP_ENTRY_LISTS_USERS``: Specifies whether the group entry in LDAP lists the users that belong to it. * ``USER_ENTRY_LISTS_GROUPS``: Default. Specifies whether the user entry in LDAP lists the groups to which the user belongs. The group attribute in LDAP is configured by the ``groupMembership`` property. |

Defining Users

Using User Distinguished Names

This approach specifies a list of templates for `userDN`. The placeholder `{0}` is replaced with the username entered by the user, and that Distinguished Name (DN) is used during LDAP bind. In the specified order, Dremio attempts to bind to the provided `userDN`. In the DN-based approach, the `baseDN`, `searchScope`, and `id` properties cannot be specified under `userAttributes`.

userDNs example

```
"userDNs": ["cn={0},dc=staticsecurity,dc=dremio,dc=com"],
  "userAttributes": {
    "firstname": "givenName",
    "lastname": "sn",
    "email": "mail"
  }
```

Using User Attributes

In this approach, you map LDAP user attributes to Dremio user attributes. The `userDN` field must not be specified in the attribute-based approach. Do not change the value of `id` in the `ad.conf` file after you start Dremio. Changing the value can result in the loss of user privileges.

userAttributes example

```
"userAttributes": {
  "baseDNs": [
    "OU=test,OU=ad,DC=drem,DC=io"
  ],
  "searchScope": "SUB_TREE",
  "id": "sAMAccountName",
  "firstname": "givenName",
  "lastname": "sn",
  "email": "mail"
}
```

Using userFilter

The following example uses the `userFilter` property to limit access to engineering group members.

userFilter example

```
"userFilter":
"&(objectClass=user)(memberOf=cn=engineering,OU=Groups,OU=ad,DC=drem,DC=io)",
```

Defining Groups

Using Group Distinguished Names

This approach specifies a list of templates for group Distinguished Names (DNs). The placeholder `{0}` is replaced with the group name entered by the user. Dremio attempts to search for the given `groupDNs` in the specified order. The `groupAttributes` property must not be specified in the DN-based approach.

Example using Group Distinguished Names

```
"groupDNs": [ "cn={0},OU=engg,OU=test,OU=ad,DC=drem,DC=io" ]
```

Using Group Attributes

In this method, use the `groupAttributes` property to specify a list of `baseDNs` and group name IDs. These properties map LDAP group attributes to Dremio group attributes. The `baseDNs`, `searchScope`, and `id` properties are required. The `groupDNs` field must not be specified in the attribute-based approach.

groupAttributes example

```
"groupAttributes": {  
  "baseDNs": [ "dc=roles,dc=dremio,dc=com" ],  
  "searchScope": "SUB_TREE",  
  "id": "CN"  
}
```

Defining User-Group Relationships

The relationship between users and groups can be defined with one of the following methods:

Group memberships

Group lists

Group Membership Method

The group membership method implements user entries in LDAP that list the groups to which the user belongs. The user entries in LDAP are configured to list their group membership via the internal field `memberOf`.

For example,

Dan is part of the **BI** group

The BI group is part of the engineering group,

`groupMembership` property will contain only the BI group, but the `groupRecursive` property will contain the engineering group.

Example settings for groupMembership and groupRecursive properties

```
"groupMembership": "memberOf",
"groupRecursive": "transitive-memberOf",
```

To establish this user-group relationship:

Specify ``groupMembership`` property.

Specify (if applicable) the ``groupRecursive`` property.

tip

If you include the ``groupRecursive`` key, ensure the value is the correct property for recursive lookups for your LDAP implementation. If you do not specify the proper property, Dremio skips recursive lookup and finds only the group membership. If you omit the ``groupRecursive`` key-value pair from your configuration, Dremio defaults to recursive lookup.

You can also specify the ``"userGroupRelationship": "USER_ENTRY_LISTS_GROUPS"`` property-value. However, this property is optional since it is the default.

Example Group Membership Configuration

```
{
  "connectionMode": "PLAIN",
  "servers": [
    {
      "hostname": "<LDAP_HOST>",
      "port": 389
    }
  ],
  "names": {
    "bindDN": "CN=Admin,OU=Users,OU=ad,DC=drem,DC=io",
    "bindPassword": "password",
    "baseDN": "dc=dremio,dc=io",
    "userFilter":
"&(objectClass=user)(|(memberOf=CN=QA,OU=temps,OU=test,OU=ad,DC=drem,DC=io)(memberOf=CN=qa,OU=engg,OU=test,OU=ad,DC=drem,DC=io))",
    "userAttributes": {
      "baseDNs": [
        "OU=test,OU=ad,DC=drem,DC=io"
      ],
      "searchScope": "SUB_TREE",
      "id": "sAMAccountName",
      "firstname": "givenName",
      "lastname": "sn",
      "email": "mail"
    },
    "groupMembership": "memberOf",
    "groupRecursive": "transitive-memberOf",
    "groupDNs": ["cn={0},OU=engg,OU=test,OU=ad,DC=drem,DC=io"],
    "groupFilter": "(objectClass=group)",
    "autoAdminFirstUser": true
  }
}
```

Group List Method

The group list method implements user-group relationships where the group entry lists the users that belong to that group.

For example,

`uid` is the ID attribute used for the user entry, and `memberUid` is the ID attribute used for the group entry.

Dan's ID is 1234, represented by the attribute `uid` in Dan's LDAP records.

Dan is part of the **BI** group

The LDAP entry for group **BI** lists `memberUid = 1234`, indicating that Dan is a valid group member.

Example Group List

```
"userGroupRelationship": "GROUP_ENTRY_LISTS_USERS",
"groupEntryListsUsers": {
  "userEntryUserIdAttribute": "uid",
  "groupEntryUserIdAttribute": "memberUid"
}
```

To establish user-group relationships in `ad.json` using the group list method:

Set `userGroupRelationship` to `GROUP_ENTRY_LISTS_USERS`.

Specify the `groupEntryListsUsers` property and its sub-properties, `userEntryUserIdAttribute` and `groupEntryUserIdAttribute`.

Example Group List Configuration

```
{
  "connectionMode": "PLAIN",
  "servers": [
    {
      "hostname": "host_ip",
      "port": 389
    }
  ],
  "names": {
    "bindDN": "CN=admin,DC=drem,DC=io",
    "bindMethod": "UNAUTHENTICATED",
    "bindPassword": "admin",
    "baseDN": "dc=drem,dc=io",
    "userFilter": "&(objectClass=posixAccount)",
    "userAttributes": {
      "baseDNs": [
        "OU=Users,OU=ldaptest,DC=drem,DC=io",
      ],
      "id": "uid",
      "firstname": "givenName",
      "lastname": "sn",
    }
  }
}
```



```

        "email": "mail"
    },
    "userGroupRelationship": "GROUP_ENTRY_LISTS_USERS",
    "groupEntryListsUsers": {
        "userEntryUserIdAttribute": "uid",
        "groupEntryUserIdAttribute": "memberUid"
    },
    "groupDNs": [ "cn={0},OU=test,OU=ldaptest,DC=drem,DC=io",
                  "cn={0},OU=dev,OU=ldaptest,DC=drem,DC=io" ],
    "groupFilter": "(objectClass=posixGroup)(objectClass=sub)",
    "autoAdminFirstUser": false
}
}

```

Bind Password Options

Dremio offers several options for managing the bind password.

Encryption

For customers with stringent security standards and requirements, password encryption provides a secure method for communicating key information with the LDAP service. Encryption is accomplished using the CLI command ``dremio-admin encrypt``.

To encrypt the bind password, follow these steps:

Run ``dremio-admin encrypt`` as the ``dremio`` service user.

LDAP secret Encryption

```
sudo su - dremio bin/dremio-admin encrypt <yourSecret>
```

If running the command as the ``dremio`` user is impossible, change the owner and group of the ``$DREMIO_HOME/data/security`` folder and underlying files to the ``dremio`` service user.

Dremio outputs:

Example LDAP Encryption Output

```
secret:1.FxLevnDdoHx58x7VZmBpNExUiM76_u7XAXo1SJ8mCJxzeC1SirK2Jm5aBRR-h2_r8iyp0AcRYSzH4uy
P33Vg6Fh94bV6evuQ.wENZ7fgdJBw92wy4DiPhpJRzNP07wBaVpspv8KyggMfYV2en3YPFZw==
```

Copy the entire output to ``bindPassword`` in ``ad.json``.

Copy the modified ``ad.json`` file to every coordinator node in the Dremio cluster.

Other Bind Password Options

Other options are available for ``bindPassword``:

``env``: the ``bindPassword`` is set to ``env:ldap`` with the environment variable set by the command ``export ldap <secret>`` where ``<secret>`` is the output.

``file``: the ``bindPassword`` is set to ``file:///tmp/test.file`` where the file specified contains the output secret.

``data``: The secret is in base64 format. The `bindPassword` is then set to ``data:text/plain;base64,SGVsbG8sIFdvcmxklQ==``.

Options ``env`` and ``file`` apply to the local node. If you use a multi-coordinator configuration, you must do this for each coordinator node. However, this method contains the raw secret in the ``env`` scheme and file. Only ``secret`` uses an encrypted secret.

Admin Users

To specify users/groups as administrators up-front, during initial login:

In the ``ad.json`` file, set ``autoAdminFirstUser`` to false.

Example property for defining Admin users

```
"autoAdminFirstUser": false
```

Create a file called ``bootstrap-admin-users.json`` that contains ``users`` and ``groups`` arrays to specify the names of the users and groups that should belong to the ``ADMIN`` role. Use the Common Name (CN) for each user and group you list in the arrays.

Example settings for users and groups properties

```
{
  users: ["joe", "bob"],
  groups: ["marketers", "sales wizards"]
}
```

When you set ``autoAdminFirstUser`` to ``false``, then you **must** specify users/groups in a ``bootstrap-admin-users.json`` file. Otherwise, an administrator won't be specified. The users/groups specified in the ``bootstrap-admin-users.json`` file are used only during initial login and when ``autoAdminFirstUser`` is set to ``false``. To add other users or groups to the ``Admin`` role **after the initial login**, use the Dremio console.

Add the configuration to your deployment:

Kubernetes

Standalone

This can be done in one of two ways:

Method 1 (Preferred)

Add the configuration of your ``bootstrap-admin-users.json`` file to your ``values-override.yaml`` via the ``configFiles`` option. This approach is detailed in [Additional Config Files](#).

Method 2

Perform a ``helm install`` with the ``set-file "dremio.configFiles.bootstrap-admin-users\,json"=/your/local/path/here`` option, indicating the location of the ``bootstrap-admin-users.json`` file. For additional information, see [step 1 in Deploying Dremio to Kubernetes](#).

Place ``bootstrap-admin-users.json`` under the Dremio configuration directory.

Start Dremio

Was this page helpful?

[\[Previous](#)

[Identity Providers\]\(/current/security/authentication/identity-providers/\)](#)[\[Next](#)

[Microsoft](#)

[Entra](#)

[ID\]\(/current/security/authentication/identity-providers/microsoft-entra-id\)](#)

[Configuring Dremio for LDAP](#)

[LDAP Properties](#)

[LDAP Connection Mode](#)

[LDAP Server Configuration](#)

[LDAP User and Groups](#)

[Defining Users](#)

[Using User Distinguished Names](#)

[Using User Attributes](#)

[Using userFilter](#)

[Defining Groups](#)

[Using Group Distinguished Names](#)

[Using Group Attributes](#)

[Defining User-Group Relationships](#)

[Group Membership Method](#)

[Group List Method](#)

[Bind Password Options](#)

[Encryption](#)

[Other Bind Password Options](#)

[Admin Users](#)

Source: [dremio-software-v25-api.md](#)

Source:
<https://docs.dremio.com/25.x/reference/api/>

Version: 25.x

On this page

API Reference

The Dremio REST API is organized by resource types such as `sources` and is designed around RESTful principles. HTTP `GET` is used to retrieve existing resources, `POST` creates new ones, `PUT` updates them and `DELETE` removes them.

Base URL

All API URLs referenced in this documentation have the following base URL unless otherwise specified:

Base URL

```
{hostname}/api/v3
```

Versions prior to `v3` are considered internal and are subject to change without notice.

In this documentation, curly braces (`{}`) are used to indicate sections of URLs where you have to supply a value. For example:

User-supplied values in URLs

```
/api/v3/source/{id}
```

Authentication

The Dremio REST API requires authentication with either a personal access token created in the Dremio UI or an authentication token generated with your username and password. Follow the instructions in this section to learn how to authenticate to the Dremio REST API with a personal access token or an authentication token.

Personal Access Token Enterprise

Personal access tokens (PATs) are created in the Dremio UI. Follow the instructions in [Creating a PAT](#) to get a PAT to use in API requests.

API requests that use a PAT token must use the Authorization header. The header's value must specify the `Bearer` type followed by a space and the PAT: `Bearer {PAT}`. This example request to retrieve a catalog demonstrates how to use a PAT in an API

request:

Example Request Using PAT

```
curl -X GET 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer dJhGBUDAVv+9Wlsp/I/o/87Vq+omuvpC/YEy6U25S79i74KhD2W6q2sr44emKy==' \
--header 'Content-Type: application/json'
```

Authentication Token

Authentication tokens are generated from your Dremio username and password. They expire every 30 hours. When an authentication token expires, you must generate a new one. API requests that use an authentication token must use the Authorization header.

caution

API versions prior to v3 are considered internal and are subject to change without notice.

To generate an authentication token:

Send an API request to the login URL, with your Dremio username and password in the request body.

Example Request

```
curl -X POST 'http://{hostname}/apiv2/login' \
--header 'Content-Type: application/json' \
--data-raw '{
"userName": "dremio",
"password": "dremio123"
}'
```

The request returns a JSON user structure:

Example Response

```
{
  "token": "4ksrt534vk7fkq64xh55g7776b",
  "userName": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "expires": 1686578200000,
  "email": "dremio@dremio.test",
  "userId": "5a679dd5-52d7-402a-871d-7fbee3fe8007",
  "admin": true,
  "clusterId": "7468ce46-58af-4dce-a42f-4c51048968f5",
  "clusterCreatedAt": 1681311939728,
  "version": "24.0.0-main-202305040803350903-2d5579e3",
  "permissions": {
    "canUploadProfiles": true,
    "canDownloadProfiles": true,
  }
}
```

```

    "canEmailForSupport": true,
    "canChatForSupport": false,
    "canViewAllJobs": true,
    "canCreateUser": true,
    "canCreateRole": true,
    "canCreateSource": true,
    "canUploadFile": true,
    "canManageNodeActivity": true,
    "canManageEngines": true,
    "canManageQueues": true,
    "canManageEngineRouting": true,
    "canManageSupportSettings": true
  },
  "userCreatedAt": 1681311939789
}

```

note

If your password includes single or double quotes, you may need to escape the quotes in your authentication token request. The required escapes vary depending on how you send the request.

For example, if you use cURL and the password is `example'6852``, the password value should be `example\'"6852\'` in the authentication token request.

Copy the value of the token attribute in the JSON user structure. This value is called the token string.

Append the token string to `_dremio` to compose the required Authorization header for API requests: `_dremio{tokenstring}`.

This example request to retrieve a catalog demonstrates how to use the token string retrieved in step 1:

Example Request Using Authentication Token

```

curl -X GET 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: _dremio4ksrt534vk7fkq64xh55g7776b' \
--header 'Content-Type: application/json'

```

Errors

Error messages will be sent back in the response body using the following format:

Error message format

```

{
  "errorMessage": "brief error message",
  "moreInfo": "detailed error message"
}

```

Query Parameters

Dremio supports query parameters for many API endpoints. The documentation for each API lists the supported query parameters for specific endpoints, along with any default and maximum values for the query parameters for that endpoint.

pageToken Query Parameter

Use the ``pageToken`` query parameter to split large sets of results into multiple pages.

Endpoints may support the ``pageToken`` query parameter based on either a built-in maximum number of results per page or a user-specified maximum that is established with a separate query parameter. The documentation for each API lists the built-in maximum or the query parameter to use to specify a maximum, as applicable.

note

Do not change any other query parameters included in the request URL when you use ``pageToken``.

Built-in Maximum

If the endpoint has a built-in maximum number of results per page, responses automatically include a page token attribute when the response contains more results than the built-in maximum. Use the value for this token in the request URL as the ``pageToken`` value to retrieve the next page of results.

As an example, the Reflection summary endpoint supports the ``pageToken`` parameter. If the Reflection summary contains more than 50 results, the response will include the ``nextPageToken`` attribute. To retrieve the next 50 results, add ``?pageToken={nextPageToken_value}`` to the request URL:

Example Request with pageToken Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?pageToken=BhQxNjc0MjhYi030TM2LTRlYTItYTFmYi0yM2IxYWM2ZTk0NTQSA== ' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

For subsequent requests, replace the ``pageToken`` value in the request URL with the token value from the previous response. If the response does not include a token attribute and value, you have retrieved the last page of available results.

User-Specified Maximum

For endpoints that require users to specify a maximum number of results per page with a separate query parameter, responses only include a page token attribute if your initial request URL includes the separate query parameter and the response contains more results than the maximum you specify. Add the value for this token to the request URL as the ``pageToken`` value, keeping the separate query parameter as well, to retrieve

the next page of results.

Catalog API endpoints for retrieving non-filesystem [sources](#), [spaces](#), and [folders](#) by ID or path support the ``maxChildren`` query parameter for specifying the maximum number of child objects to include in each response. If the response contains more than the specified number of child objects, the response includes the ``nextPageToken`` attribute. To retrieve the next page of results, add ``?pageToken={nextPageToken_value}`` to the request URL. This example shows a request URL that uses the ``nextPageToken`` query parameter with the ``maxChildren`` query parameter set to 25:

Example Request with maxChildren and pageToken Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4?maxChildren=25&pageToken=BhQxNjc0MjhlYi030TM2LTRlYTItYTfmYi0yM2IxYWM2ZTk0NTQSA== ' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

For subsequent requests, replace the ``pageToken`` value in the request URL with the token value from the previous response. If the response does not include a token attribute and value, you have retrieved the last page of available results.

maxResults Query Parameter

Use the ``maxResults`` query parameter to specify the maximum number of results to retrieve in each request.

For example, if you want to retrieve no more than 25 results for an endpoint that supports the ``maxResults`` query parameter, append ``?maxResults=25`` to the request URL:

Example Request with maxResults Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?maxResults=25' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

filter Query Parameter

Use the ``filter`` query parameter to filter responses so that they include only results with the specified attributes and values. The value for the ``filter`` query parameter is a URL-encoded JSON string that represents a JSON object that specifies the desired attributes and values.

As an example, the [Reflection summary](#) endpoint supports the ``filter`` query parameter for certain specific attributes. To retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with ``samples.dremio.com`` in their paths, the ``filter`` JSON object would look like this:

Example JSON Object for Filter

```
{
```



```

"reflectionType": ["RAW"],
"refreshStatus": ["MANUAL", "SCHEDULED"],
"enabledFlag": true,
"reflectionNameOrDatasetPath": "samples.dremio.com"
}

```

To use the JSON object as the `filter` value, convert it to URL-encoded JSON and add it to the request URL:

Example Request with filter Query Parameter

```

curl -X GET 'https://{hostname}/api/v3/reflection-summary?filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'

```

Read the endpoint-specific documentation to learn which attributes each endpoint supports for the `filter` query parameter.

orderBy Query Parameter

Use the `orderBy` query parameter to organize the response in ascending or descending order based on the value of the specified attribute. The default is ascending order. To specify descending order, add a `-` character before the attribute name.

For example, the [Reflection summary](#) endpoint supports ordering the response by reflectionName, datasetName, or reflectionType. To organize the response in ascending order by reflectionName:

Example Request with orderBy Query Parameter (Ascending Order)

```

curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'

```

To organize the response in descending order, add a `-` before the attribute name:

Example Request with orderBy Query Parameter (Descending Order)

```

curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=-reflectionName' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'

```

Read the endpoint-specific documentation to learn which attributes each endpoint supports for the `orderBy` query parameter.

limit and offset Query Parameters

The ``limit`` query parameter allows you to retrieve a specific number of results. For endpoints that support the ``limit`` query parameter, you can specify the number of results to retrieve. For example, if you only want to retrieve the first 10 available results, add ``?limit=10`` to the request URL:

Example Request for First 10 Results with Limit Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=10' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

The ``offset`` query parameter allows you to skip a specific number of results in the response. When a response contains many results, you can use the ``limit`` and ``offset`` query parameters together to break the response into pages.

For example, consider a job result response object that contains 5000 results. The Job API allows you to retrieve a maximum of 500 results per request. To retrieve all 5000 results, start by adding ``?limit=500`` to the request URL to retrieve the first 500:

Example Request for First 500 Results with Limit Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

In the next request, to retrieve the next 500 results (rows 501-1000), add ``&offset=500`` to the request URL:

Example Request for Results 501-1000 with limit and offset Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500&offset=500' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

To retrieve the next 500 results (rows 1001-1500), increment the ``offset`` parameter to 1000 in the next request:

Example Request for Results 1001-1500 with limit and offset Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500&offset=1000' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Continue incrementing the ``offset`` parameter in requests until you have retrieved all 5000 results.

Read the documentation for each API to learn about endpoint-specific support for the ``limit`` and ``offset`` query parameters.

type Query Parameter

Use the `type` query parameter to limit your request so that the response includes only results for the type you specify.

For example, if an endpoint supports the `type` query parameter, and the endpoint's list of valid values includes `SOURCE`, you can limit the response so that it includes only results for sources. Append `?type=SOURCE` to the request URL:

Example Request with type Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/privileges?type=SOURCE' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `type` query parameter, including lists of valid values.

include and exclude Query Parameters

Some APIs exclude non-default attributes or include lengthy attributes in the default GET responses. These APIs support the `include` and `exclude` query parameters, which you can use to include or exclude certain attributes in the responses for GET endpoints.

The `include` query parameter allows you to include non-default attributes in the response. For example, in the Catalog API, you can include a catalog object's `permissions` array in the response:

Example Request with include Query Parameter

```
curl -X GET
'https://{hostname}/api/v3/catalog/ffbe8c1d-1db7-48d1-9c58-f452838fedc0?include=permissions' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

The `exclude` query parameter allows you to exclude supported attributes from the response. For example, this Catalog API request excludes the object's `children` attribute from the response:

Example Request with exclude Query Parameter

```
curl -X GET
'https://{hostname}/api/v3/catalog/ffbe8c1d-1db7-48d1-9c58-f452838fedc0?exclude=children' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `include` and `exclude` query parameters, including information about supported values.

createdBy Query Parameter

Use the `createdBy` query parameter to limit the request to objects created by a specific user.

For example, this Scripts API request retrieves only scripts that were created by the user whose ID is `8be516f3-04c4-4d19-824d-5a70b3c4442e`:

Example Request with createdBy Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?createdBy=8be516f3-04c4-4d19-824d-5a70b3c4442e' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

ownedBy Query Parameter

Use the `ownedBy` query parameter to limit the request to objects owned by a specific user.

For example, this Scripts API request retrieves only scripts that are owned by the user whose ID is `8be516f3-04c4-4d19-824d-5a70b3c4442e`:

Example Request with ownedBy Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?ownedBy=8be516f3-04c4-4d19-824d-5a70b3c4442e' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

search Query Parameter

Use the `search` query parameter to limit the request to objects that contain values that include the search string.

For example, the Scripts API supports the `search` query parameter for the name attribute. This Scripts API request retrieves only scripts whose values for the name attribute include `dev`:

Example Request with search Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?search=dev' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `search` query parameter, including information about searchable attributes.

maxChildren Query Parameter

The ``maxChildren`` query parameter allows you to specify the maximum number of child objects to include in each response. This example shows a request URL that uses the ``nextPageToken`` query parameter with the ``maxChildren`` query parameter set to 25:

Example Request with maxChildren Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4?maxChildren=25' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Use the ``maxChildren`` query parameter in concert with the ``pageToken`` query parameter to split large sets of results into multiple pages.

Response Headers

Dremio API responses include HTTP headers that provide additional information about responses. Each header includes a case-insensitive name and a value, separated by a colon.

The following response headers are common to Dremio API endpoints:

Header Name	Example Value	Description
Allow	GET,OPTIONS	Request methods the endpoint supports.
Cache-Control	no-cache, no-store	Caching instructions for browsers and shared caches.
Content-Length	2188	Size of the response body, in bytes.
Content-Type	application/json	MIME type of the object.
Date	Fri, 14 Apr 2023 19:39:53 GMT	Date and time when the response originated.
Vary	Accept-Encoding, User-Agent	Names of request headers that could have affected the response's generation.
x-content-type-options	nosniff	Instructions about following the MIME type in the content-type header. Blocks content sniffing.
x-xss-protection	1; mode=block	Instructions used to stop pages from loading when a browser detects reflected cross-site scripting attacks.

Was this page helpful?

[Previous

Reference](/25.x/reference/)[Next

Catalog](/25.x/reference/api/catalog/)

Base URL

Authentication

Personal Access Token Enterprise

Authentication Token

Errors

Query Parameters

pageToken Query Parameter

maxResults Query Parameter

filter Query Parameter

orderBy Query Parameter

limit and offset Query Parameters

type Query Parameter

include and exclude Query Parameters

createdBy Query Parameter

ownedBy Query Parameter

search Query Parameter

maxChildren Query Parameter

Response Headers

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/>

Version: 25.x

On this page

Catalog

Use the Catalog API to retrieve and manage [sources](#) and [spaces](#) as well as the [folders](#), [files](#), [tables](#), and [views](#) they contain. The Catalog API also includes endpoints for retrieving [lineage](#) information for datasets and for creating and managing [tags](#), [wikis](#), [privileges](#), and [grants](#) on catalog objects.

Use the Catalog API endpoint described on this page to retrieve a list of the spaces and sources in your Dremio organization. The response contains the IDs required to make requests to other Catalog API endpoints to create, retrieve, update, and delete objects in your catalog.

Catalog Object

```
{
  "data": [
    {
      "id": "a7b1bc39-bffa-4c30-a5eb-5bdaf5bd0959",
      "path": [
        "@dremio"
      ]
    }
  ]
}
```

```

],
"tag": "0QVA7wGyiY0=",
"type": "CONTAINER",
"containerType": "HOME",
"stats": {
  "datasetCount": 18,
  "datasetCountBounded": false
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",
  "TRUNCATE",
  "DELETE",
  "UPDATE",
  "EXECUTE",
  "CREATE_SOURCE",
  "ALL"
]
},
{
  "id": "ed1013cb-4fea-6552-8d43-015215a38bcc",
  "path": [
    "Testing"
  ],
  "tag": "PR1M7B1Rhjs=",
  "type": "CONTAINER",
  "containerType": "SPACE",
  "stats": {
    "datasetCount": 3,
    "datasetCountBounded": false
  },
  "createdAt": "2023-02-14T19:28:40.840Z",
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",

```

```

        "INSERT",
        "TRUNCATE",
        "DELETE",
        "UPDATE",
        "EXECUTE",
        "CREATE_SOURCE",
        "ALL"
    ]
},
{
    "id": "6b714877-760e-115b-aeef-799430b3ceab",
    "path": [
        "Samples"
    ],
    "tag": "nEjWZGnrA00=",
    "type": "CONTAINER",
    "containerType": "SOURCE",
    "stats": {
        "datasetCount": 10,
        "datasetCountBounded": false
    },
    "createdAt": "2023-01-04T22:13:02.536Z",
    "permissions": [
        "READ",
        "WRITE",
        "ALTER_REFLECTION",
        "SELECT",
        "ALTER",
        "VIEW_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "CREATE_TABLE",
        "DROP",
        "EXTERNAL_QUERY",
        "INSERT",
        "TRUNCATE",
        "DELETE",
        "UPDATE",
        "EXECUTE",
        "CREATE_SOURCE",
        "ALL"
    ]
}
]
}
}

```

Catalog Attributes

data Array of Object

List of catalog objects in the Dremio organization.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the catalog object.

Example: ed1013cb-4fea-6552-8d43-015215a38bcc

path Array of String

Path of the catalog object within Dremio, expressed as an array.

Example: ["Testing"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the catalog object.

Example: PR1M7B1Rhjs=

type String

Type of the catalog object. For objects that can contain other catalog objects (the only objects this endpoint retrieves), the type is `CONTAINER`.

Example: CONTAINER

containerType String

For catalog objects with the type CONTAINER, the type of container.

Enum: SPACE, SOURCE, FOLDER, HOME

Example: SPACE

stats Object

Information about the number of datasets in the catalog object and whether the dataset count is bounded. Appears in the response only if the request URL includes the datasetCount query parameter.

Example: {"datasetCount": 18,"datasetCountBounded": false}

createdAt String

Date and time that the catalog object was created, in UTC format.

Example: 2023-02-14T19:28:40.840Z

permissions Array of String

Enterprise-only. List of the privileges that you have on the catalog object. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read Privileges.

Example:

```
["READ","WRITE","ALTER\\_REFLECTION","SELECT","ALTER","VIEW\\_REFLECTION","MODIFY","MANAGE\\_GRANTS","CREATE\\_TABLE","DROP","EXTERNAL\\_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE\\_SOURCE","ALL"]
```

Attributes of the `stats` Object

datasetCount Integer

Number of datasets the catalog object contains.

Example: 18

datasetCountBounded Boolean

If the dataset count is bounded, the value is `true`. Otherwise, the value is `false`.

Example: false

Retrieving a Catalog

Retrieve the catalog for the current Dremio instance.

Method and URL

```
GET /api/v3/catalog
```

Parameters

include Query String Optional

Include a non-default attribute in the response. The available values for the include query parameter are `permissions` (Enterprise-only) and `datasetCount`. Specify `permissions` to include each catalog object's permissions array in the response. Specify `datasetCount` to include the stats object in the response. For more information, read include and exclude Query Parameters.

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "id": "a7b1bc39-bffa-4c30-a5eb-5bdaf5bd0959",
      "path": [
        "@dremio"
```

```

    ],
    "tag": "0QVA7wGyiY0=",
    "type": "CONTAINER",
    "containerType": "HOME"
  },
  {
    "id": "ed1013cb-4fea-6552-8d43-015215a38bcc",
    "path": [
      "Testing"
    ],
    "tag": "PR1M7B1Rhjs=",
    "type": "CONTAINER",
    "containerType": "SPACE",
    "createdAt": "2023-02-14T19:28:40.840Z"
  },
  {
    "id": "6b714877-760e-115b-aefd-799430b3ceab",
    "path": [
      "Samples"
    ],
    "tag": "nEjWZGnrA00=",
    "type": "CONTAINER",
    "containerType": "SOURCE",
    "createdAt": "2023-01-04T22:13:02.536Z"
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

API Reference](/25.x/reference/api/)[Next

Source](/25.x/reference/api/catalog/source/)

Catalog Attributes

Retrieving a Catalog

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/datasets/>

Version: 25.x

On this page

Dataset

note

The Dataset API is supported in Dremio 25.0.5+.

Use the Dataset API to retrieve Dremio's Reflection recommendations for your datasets.

Dataset Object (All Reflections)

```
{
  "data": [
    {
      "type": "RAW",
      "enabled": true,
      "arrowCachingEnabled": false,
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {
          "name": "fare_amount"
        },
        {
          "name": "tip_amount"
        },
        {
          "name": "total_amount"
        }
      ],
      "partitionFields": [
        {
          "name": "dropoff_date"
        },
        {
          "name": "passenger_count"
        }
      ],
      "entityType": "reflection"
    },
  ],
}
```

```

{
  "type": "AGGREGATION",
  "enabled": true,
  "arrowCachingEnabled": false,
  "dimensionFields": [
    {
      "name": "passenger_count",
      "granularity": "DATE"
    }
  ],
  "measureFields": [
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "entityType": "reflection"
}
]
}

```

Dataset Attributes

data Array of Object

List of recommended Reflection objects for the specified dataset ID.

Attributes of objects in the `data` Array

type String

Reflection type. For details, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: RAW

enabled Boolean

If the Reflection is available for accelerating queries, `true`. Otherwise, `false`.

Example: true

arrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, `true`. Otherwise, `false`.

Example: false

displayFields Array of Object

Information about the fields displayed from the anchor dataset. Each object in the displayFields array contains one attribute: name. Included only for raw Reflections. Not included for aggregation Reflections.

Example:

```
[{"name":"pickup_datetime"}, {"name":"passenger_count"}, {"name":"trip_distance_mi"}, {"name":"fare_amount"}, {"name":"tip_amount"}, {"name":"total_amount"}]
```

dimensionFields Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each object in the dimensionFields array contains two attributes: name and granularity. Included only for aggregation Reflections. If the anchor dataset does not include any dimension fields, the dimensionFields value is an empty array. Not included for raw Reflections.

Example: [{"name":"passenger_count","granularity":"DATE"}]

measureFields Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each object in the measureFields array contains two attributes: name and measureTypeList. Included only for aggregation Reflections. If the anchor dataset does not include any measure fields, the measureFields value is an empty array. Not

included for raw Reflections.

Example:

```
[{"name":"total\_amount","measureTypeList":["COUNT","SUM"]}, {"name":"trip\_distance\_mi","measureTypeList":["COUNT","SUM"]}, {"name":"fare\_amount","measureTypeList":["COUNT","SUM"]}, {"name":"tip\_amount","measureTypeList":["COUNT","SUM"]}]
```

partitionFields Array of Object

Information about the fields from the anchor dataset used to partition data in the Reflection. Each object in the partitionFields array contains one attribute: name. Included only for aggregation Reflections. If the anchor dataset does not include any partition fields, the partitionFields value is an empty array. Not included for raw Reflections.

Example: [{"name": "dropoff_date"}, {"name": "passenger_count"}]

entityType String

Type of entity. For objects in dataset responses, the entityType is `reflection`.

Creating and Retrieving Reflection Recommendations for a Dataset

Create Reflection recommendations for the specified dataset. The response contains the Reflection recommendations.

Method and URL

```
POST /api/v3/dataset/{id}/reflection/recommendation/{type}/
```

Parameters

id Path String (UUID)

The id of the dataset for which you want to create and retrieve recommended Reflections.

Example: 88e5fbdf-4b56-4286-9b8b-bb48e1f350eb

type Path String

The type of Reflection recommendations you want to create and retrieve.

ALL: Create and retrieve both raw and aggregation Reflection recommendations.

RAW: Create and retrieve only raw Reflection recommendations.

AGG: Create and retrieve only aggregation Reflection recommendations.

NOTE: The type is not case-sensitive. For example, `AGG`, `agg`, and `aGg` are valid type values for aggregation Reflection recommendations.

Example: ALL

Example Request (All Reflections)

```
curl -X POST 'https://{hostname}/api/v3/dataset/88e5fbdf-4b56-4286-9b8b-bb48e1f350eb/reflection/recommendation/ALL/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response (All Reflections)

```
{
  "data": [
    {
      "type": "RAW",
      "enabled": true,
      "arrowCachingEnabled": false,
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {
          "name": "fare_amount"
        },
        {
          "name": "tip_amount"
        },
        {
          "name": "total_amount"
        }
      ],
      "partitionFields": [
        {
          "name": "dropoff_date"
        },
        {
          "name": "passenger_count"
        }
      ],
      "entityType": "reflection"
    },
    {
      "type": "AGGREGATION",
      "enabled": true,
      "arrowCachingEnabled": false,
      "dimensionFields": [
        {
          "name": "passenger_count",
          "granularity": "DATE"
        }
      ]
    }
  ]
}
```



```

    }
  ],
  "measureFields": [
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "entityType": "reflection"
}
]
}

```

Example Request (Raw Reflections)

```

curl -X POST \
  'https://{hostname}/api/v3/dataset/88e5fbdf-4b56-4286-9b8b-bb48e1f350eb/reflection/recom
mendation/RAW/' \
  --header 'Authorization: Bearer <PersonalAccessToken>' \
  --header 'Content-Type: application/json'

```

Example Response (Raw Reflections)

```
{
  "data": [
    {
      "type": "RAW",
      "enabled": true,
      "arrowCachingEnabled": false,
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {
          "name": "fare_amount"
        },
        {
          "name": "tip_amount"
        },
        {
          "name": "total_amount"
        }
      ],
      "partitionFields": [
        {
          "name": "dropoff_date"
        },
        {
          "name": "passenger_count"
        }
      ],
      "entityType": "reflection"
    }
  ]
}
```

Example Request (Aggregation Reflections)

```
curl -X POST 'https://{hostname}/api/v3/dataset/88e5fbdf-4b56-4286-9b8b-bb48e1f350eb/reflection/recommendation/AGG/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response (Aggregation Reflections)

```
{
  "data": [
```

```

{
  "type": "AGGREGATION",
  "enabled": true,
  "arrowCachingEnabled": false,
  "dimensionFields": [
    {
      "name": "passenger_count",
      "granularity": "DATE"
    }
  ],
  "measureFields": [
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "entityType": "reflection"
}
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

405 Method Not Allowed

500 Internal Server Error

Was this page helpful?

[Previous

Grants]([25.x/reference/api/catalog/grants])[Next

External Token Providers]([25.x/reference/api/external-token-providers/)

Dataset Attributes

Creating and Retrieving Reflection Recommendations for a Dataset

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/external-token-providers/>

Version: 25.x

On this page

External Token Providers Enterprise

Use the External Token Providers API to manage external token providers that enable client applications to use a JSON Web Token (JWT) issued by an identity provider to authenticate to Dremio.

External Token Providers Object

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "userClaim": "upn",
  "issuer":
    "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks":
    "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/k"
```

```
    "type": "JWT",  
    "state": "ENABLED"  
  }  
}
```

External Token Providers Attributes

id String (UUID)

Unique identifier of the external token provider. Generated by Dremio and immutable.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

name String

User-provided name of the external token provider. Used for display only.

Example: My Token Provider

audience Array of String

Intended recipients of the JSON Web Token (JWT). If there is only one audience for the JWT, then the audience value contains only one string.

Example: ["f7dd9e0-8332-4131-95ce-b350c3bbeab2"]

userClaim String

Key name for the target claim in the JSON Web Token (JWT). The target claim's value corresponds to the Dremio username.

Example: upn

issuer String

URL that identifies the principal that issued the JSON Web Token (JWT).

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0>

jwt String

Endpoint that hosts the JWK Set (JWKS), a set of public keys used to verify the JSON Web Token (JWT) signature.

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys>

type String

Type of the tokens associated with the external token provider. The type is always `JWT`.

state String

Current state of the external token provider: `ENABLED` (default) or `DISABLED`.

Example: ENABLED

Listing All External Token Providers

Retrieve a list of all available external token providers and the metadata for each provider.

Method and URL

```
GET /api/v3/external-token-providers/
```

Parameters

pageToken Query Query Optional

Token for retrieving the next page of external token providers. If the Dremio instance has more providers than the maximum per page (default 5), the response will include a nextPageToken after the data array. Use the nextPageToken value in your request URL as the pageToken value. Do not change any other query parameters included in the request URL when you use pageToken. Read [pageToken Query Parameter](#) for usage examples.

limit Query Integer Optional

Number of rows to return. Maximum valid value is `99`. Default is `5`. Read [Limit and Offset Query Parameters](#) for usage examples.

Example Request

```
curl -X GET 'https://{hostname}/api/v3/external-token-providers' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "id": "d21bbf72-6ab7-45e8-9466-ae1d9ffe92a7",
      "name": "My First Provider",
      "type": "JWT",
      "state": "DISABLED"
    },
    {
      "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
      "name": "My Token Provider",
      "type": "JWT",
      "state": "ENABLED"
    }
  ]
}
```

```
}
```

Response Status Codes

200 OK

403 Forbidden

Creating an External Token Provider

Create an external token provider.

Method and URL

```
POST /api/v3/external-token-providers
```

Parameters

name Body String

Name to use for the external token provider.

Example: My Token Provider

audience Body Array of String

Intended recipients of the JSON Web Token (JWT).

Example: ["f7fdd9e0-8332-4131-95ce-b350c3bbeab2"]

userClaim Body String

Key name for the target claim in the JSON Web Token (JWT). The target claim's value corresponds to the Dremio username.

Example: upn

issuer Body String

URL that identifies the principal that issued the JSON Web Token (JWT).

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0>

jwt Body String Optional

Endpoint that hosts the JWK Set (JWKS), a set of public keys used to verify the JSON Web Token (JWT) signature. If you do not provide a jwt value, Dremio retrieves the value from ``<issuer>/well-known/openid-configuration``.

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discover

y/v2.0/keys>

Example Request

```
curl -X POST 'https://{hostname}/api/v3/external-token-providers' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "issuer":
"https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/v2.0",
  "jwks":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/k
eys",
  "userClaim": "upn"
}'
```

Example Response

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "userClaim": "upn",
  "issuer":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/k
eys",
  "type": "JWT",
  "state": "ENABLED"
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

Retrieving an External Token Provider by ID

Retrieve a specific external token provider by the providers's ID.

Method and URL

GET /api/v3/external-token-providers/{id}

Parameters

id Path String (UUID)

Unique identifier of the external token provider.

Example:a32191f2-ed6-4533-9a17-1532eea015aa

Example Request

```
curl -X GET 'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "userClaim": "upn",
  "issuer": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
  "type": "JWT",
  "state": "ENABLED"
}
```

Response Status Codes

200 OK

403 Forbidden

404 Not Found

Updating an External Token Provider

Update the specified external token provider.

Method and URL

Parameters

id Path String (UUID)

Unique identifier of the external token provider.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

name Body String

User-provided name of the external token provider.

Example: My Token Provider

audience Body Array of String

Intended recipients of the JSON Web Token (JWT). If there is only one audience for the JWT, then the audience value contains only one string.

Example: ["28edee01-4d0d-46ed-b1ae-52139bc3b3ad"]

userClaim Body String

Key name for the target claim in the JSON Web Token (JWT). The target claim's value corresponds to the Dremio username.

Example: preferred_username

issuer Body String

URL that identifies the principal that issued the JSON Web Token (JWT).

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0>

jwt Body String Optional

Endpoint that hosts the JWK Set (JWKS), a set of public keys used to verify the JSON Web Token (JWT) signature. If you do not provide a jwt value, Dremio retrieves the value from `

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys>

type Body String Optional

Type of the tokens associated with the external token provider. The type is always `JWT`.

Example: JWT

state Body String Optional

Current state of the external token provider: `ENABLED` or `DISABLED`. If the update request does not include the state parameter, Dremio does not change the state.

Example: ENABLED

Example Request

```
curl -X PUT \
'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "My Token Provider",
  "audience": [
    "28edee01-4d0d-46ed-b1ae-52139bc3b3ad"
  ],
  "userClaim": "preferred_username",
  "issuer":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0"
}'
```

Example Response

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "28edee01-4d0d-46ed-b1ae-52139bc3b3ad"
  ],
  "userClaim": "preferred_username",
  "issuer":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
  "type": "JWT",
  "state": "ENABLED"
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Updating an External Token Provider's State

Update the state for the specified external token provider.

Method and URL

PATCH /api/v3/external-token-providers/{id}/state

Parameters

id Path String (UUID)

Unique identifier of the external token provider.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

state Body String

Current state of the external token provider: `ENABLED` or `DISABLED`.

Example: DISABLED

Example Request

```
curl -X PUT \
'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa/state' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "state": "DISABLED"
}'
```

A successful request to update the state for an external token provider returns an empty response with the HTTP `204 No Content` status response code.

Response Status Codes

204 No Content

400 Bad Request

403 Forbidden

404 Not Found

Deleting an External Token Provider

Delete the specified external token provider.

Method and URL

DELETE /api/v3/external-token-providers/{id}

Parameters

id Path String (UUID)

Unique identifier of the external token provider that you want to delete.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

403 Forbidden

Was this page helpful?

[Previous

Dataset]/(25.x/reference/api/datasets/)[Next

Job]/(25.x/reference/api/job/)

External Token Providers Attributes

Listing All External Token Providers

Parameters

Response Status Codes

Creating an External Token Provider

Parameters

Response Status Codes

Retrieving an External Token Provider by ID

Parameters

Response Status Codes

Updating an External Token Provider

Parameters

Response Status Codes

Updating an External Token Provider's State

Parameters

Response Status Codes

Deleting an External Token Provider

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/job/>

Version: 25.x

On this page

Job

Use the Job API to get information about a specific job and cancel a running job.

note

To retrieve results for a specific job, use the [Job Results](#) endpoint.

Job Object

```
{
  "jobState": "COMPLETED",
  "rowCount": 1003904,
  "errorMessage": "",
  "startedAt": "2022-12-13T18:34:14.069Z",
  "endedAt": "2022-12-13T18:35:09.963Z",
  "acceleration": {
    "reflectionRelationships": [
      {
        "datasetId": "ef99ab32-aa47-4f4c-4d1c-d40f8035b846",
        "reflectionId": "63fd1c83-5cde-4133-9e2d-60543550580a",
        "relationship": "CONSIDERED"
      },
      {
        "datasetId": "596c489c-7949-485b-92a9-c32a4cb51fa2",
        "reflectionId": "65747723-2319-430p-8a36-3d40b26f45ae",
        "relationship": "MATCHED"
      }
    ]
  },
  "queryType": "UI_RUN",
  "queueName": "LARGE",
}
```

```
"queueId": "f6a5ef4f-ce5c-4be4-95b2-092c36721dc5",  
"resourceSchedulingStartedAt": "2022-12-13T18:34:14.977Z",  
"resourceSchedulingEndedAt": "2022-12-13T18:34:14.995Z",  
"cancellationReason": ""  
}
```

Job Attributes

jobState String

The job's status. Values `COMPLETED`, `CANCELED`, and `FAILED` are final; other values are considered in running state.

Enum: NOT_SUBMITTED, STARTING, RUNNING, COMPLETED, CANCELED, FAILED, CANCELLATION_REQUESTED, PLANNING, PENDING, METADATA_RETRIEVAL, QUEUED, ENGINE_START, EXECUTION_PLANNING, INVALID_STATE

Example: COMPLETED

rowCount Integer

For jobs with `COMPLETED` jobState, the number of rows the job returned. If jobState is not `COMPLETED`, rowCount value is `0`.

Example: 11

errorMessage String

For jobs with `FAILED` jobState, the error that caused the failure. For all other jobs, the errorMessage value is empty.

Example: Column 'user_id' not found in any table.

startedAt String

Date and time when the job started, in UTC format.

Example: 2022-12-09T20:16:15.694Z

endedAt String

Date and time when the job ended, in UTC format.

Example: 2022-12-09T20:16:19.939Z

acceleration Object

For jobs with applicable Reflections, provides more information about the Reflections and their relationships to the job. For jobs that do not have applicable Reflections, the response does not include the acceleration object.

queryType String

Job type. If the job's queryType is not set, the value is `UNKNOWN`.

Enum: UI_RUN, UI_PREVIEW, UI_INTERNAL_PREVIEW, UI_INTERNAL_RUN,

UI_EXPORT, ODBC, JDBC, REST, ACCELERATOR_CREATE, ACCELERATOR_DROP, UNKNOWN, PREPARE_INTERNAL, ACCELERATOR_EXPLAIN, UI_INITIAL_PREVIEW

Example: UI_RUN

queueName String

Name of the workload management (WLM) queue to which the job was routed.

Example: SMALL

queueId String

ID of the workload management (WLM) queue to which the job was routed.

Example: f6a5ef4f-ce5c-4be4-95b2-092c36721dc5

resourceSchedulingStartedAt String

Date and time when the Dremio engine started scheduling the job.

Example: 2022-12-09T20:16:16.141Z

resourceSchedulingEndedAt String

Date and time when Dremio engine scheduling ended for the job.

Example: 2022-12-09T20:16:16.162Z

cancellationReason String

For canceled jobs, the reason for the cancellation. For all other jobs, the cancellationReason value is empty.

Example: Query was cancelled due to low memory.

Attributes of the `acceleration` Object

reflectionRelationships Array of Object

Information about the dataset, Reflection, and type of relationship for each applicable Reflection.

Attributes of Objects in the `reflectionRelationships` Array

datasetId String (UUID)

Unique identifier of the dataset associated with the Reflection.

Example: 596c489c-7949-485b-92a9-c32a4cb51fa2

reflectionId String (UUID)

Unique identifier of the Reflection.

Example: 65747723-2319-430p-8a36-3d40b26f45ae

relationship String

Type of relationship between the Reflection and the job.

Enum: CONSIDERED, MATCHED, CHOSEN

Example: MATCHED

Retrieving a Job

Retrieve the specified job.

Method and URL

```
GET /api/v3/job/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the job to retrieve. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

Example Request

```
curl -X GET 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response for a COMPLETED Job

```
{
  "jobState": "COMPLETED",
  "rowCount": 1003904,
  "errorMessage": "",
  "startedAt": "2022-12-13T18:34:14.069Z",
  "endedAt": "2022-12-13T18:35:09.963Z",
  "acceleration": {
    "reflectionRelationships": [
      {
        "datasetId": "ef99ab32-aa47-4f4c-4d1c-d40f8035b846",
        "reflectionId": "63fd1c83-2319-5962-8a36-60543550580a",
        "relationship": "CONSIDERED"
      },
      {
        "datasetId": "596c489c-7949-485b-92a9-c32a4cb51fa2",
        "reflectionId": "65747723-4133-9e2d-3k86-3d40b26f45ae",
        "relationship": "MATCHED"
      }
    ]
  }
},
```

```

"queryType": "UI_RUN",
"queueName": "LARGE",
"queueId": "f6a5ef4f-ce5c-4be4-95b2-092c36721dc5",
"resourceSchedulingStartedAt": "2022-12-13T18:34:14.977Z",
"resourceSchedulingEndedAt": "2022-12-13T18:34:14.995Z",
"cancellationReason": ""
}

```

Example Response for a CANCELED Job

```

{
  "jobState": "CANCELED",
  "rowCount": 0,
  "errorMessage": "",
  "startedAt": "2023-02-01T15:07:16.165Z",
  "endedAt": "2023-02-01T15:07:18.691Z",
  "queryType": "UI_RUN",
  "queueName": "LARGE",
  "queueId": "6ed7841e-e446-4536-8d47-361508e78c18",
  "resourceSchedulingStartedAt": "2023-02-01T15:07:17.124Z",
  "resourceSchedulingEndedAt": "2023-02-01T15:07:17.140Z",
  "cancellationReason": "Query cancelled by user 'USERNAME'"
}

```

Example Response for a FAILED Job

```

{
  "jobState": "FAILED",
  "rowCount": 0,
  "errorMessage": "ExecutionSetupException: One or more nodes lost connectivity during query. Identified nodes were [automaster-2.c.dremio-1093.internal:0].",
  "startedAt": "2023-02-01T16:36:35.897Z",
  "endedAt": "2023-02-01T16:37:36.098Z",
  "queryType": "UI_RUN",
  "queueName": "LARGE",
  "queueId": "3d04235f-3610-4dd3-95b6-6a29542eb600",
  "resourceSchedulingStartedAt": "2023-02-01T16:36:37.389Z",
  "resourceSchedulingEndedAt": "2023-02-01T16:36:37.437Z",
  "cancellationReason": ""
}

```

Example Response for a RUNNING Job

```

{
  "jobState": "RUNNING",
  "rowCount": 2682474,
  "errorMessage": "",
  "startedAt": "2023-02-01T21:30:10.755Z",
  "queryType": "ACCELERATOR_CREATE",
  "queueName": "LARGE",
  "queueId": "f64ff0a0-a925-4dc9-be60-e0703ce3aa24",
  "resourceSchedulingStartedAt": "2023-02-01T21:30:11.743Z",

```

```
"resourceSchedulingEndedAt": "2023-02-01T21:30:11.798Z",  
"cancellationReason": ""  
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Canceling a Running Job

Cancel the specified running job.

note

Canceling a job does not delete the job object. You can still retrieve job objects for canceled jobs.

Method and URL

```
POST /api/v3/job/{id}/cancel
```

Parameters

id Path String (UUID)

Unique identifier of the job to cancel. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

Example request

```
curl -X POST 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00/cancel'  
\  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example response

No response

Response Status Codes

204 No Content

400 Bad Request

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

External Token Providers](/25.x/reference/api/external-token-providers/)[Next

Job Results](/25.x/reference/api/job/job-results)

Job Attributes

Retrieving a Job

Parameters

Response Status Codes

Canceling a Running Job

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/ldap-authorization>

Version: 25.x

On this page

LDAP User Cache

Use the LDAP User Cache API to invalidate the authorization for all users and groups or a single user or group in the LDAP user cache.

note

You must be a member of the Dremio ADMIN role to send requests to the LDAP User Cache API.

Invalidating LDAP Authorization for All Users and Groups

Invalidate all users' and groups' LDAP authorizations.

Method and URL

```
DELETE /api/v3/cache/authorization
```

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/cache/authorization' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Invalidating LDAP Authorization for a Single User or Group by ID

Invalidate LDAP authorization for a single user or group by specifying the ID for the user or group.

note

If you do not provide the ID of a user or group in the request URL, Dremio invalidates the LDAP authorization for all users and groups in the cache.

Method and URL

```
DELETE /api/v3/cache/authorization/{id}
```

Parameters

name Path String (UUID)

Unique identifier of the Dremio user or group whose LDAP authorization you want to invalidate.

Example: 2k8bdk96-b267-4d56-9154-e48v5884h5i8

Example Request

```
curl -X DELETE \  
'https://{hostname}/api/v3/cache/authorization/2k8bdk96-b267-4d56-9154-e48v5884h5i8' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Invalidating LDAP Authorization for a Single User or Group by Name

Invalidate LDAP authorization for a single user or group by specifying the name for the user or group.

note

If you do not provide the name of a user or group in the request URL, Dremio invalidates the LDAP authorization for all users and groups in the cache.

Method and URL

```
DELETE /api/v3/cache/authorization/{name}
```

Parameters

name Path String

Name for the Dremio user or group whose LDAP authorization you want to invalidate.

Example: exampleuser1

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/cache/authorization/exampleuser1' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

Job Results]([/25.x/reference/api/job/job-results])[Next

Node Collections]([/25.x/reference/api/nodeCollections/)

Invalidating LDAP Authorization for All Users and Groups

Response Status Codes

Invalidating LDAP Authorization for a Single User or Group by ID

Parameters

Response Status Codes

Invalidating LDAP Authorization for a Single User or Group by Name

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/nodeCollections/>

Version: 25.x

On this page

Node Collections

Use the Node Collections API to deny access to Dremio cluster nodes and retrieve the currently denied nodes for the Dremio instance.

The Node Collections API does not have a corresponding Node Collection object. The endpoints return a list of the currently denied nodes in the response.

note

You must be a member of the Dremio ADMIN role to send requests to the Node Collections API.

Denying Nodes

Deny access to the specified Dremio cluster nodes.

Method and URL

```
POST /api/v3/nodeCollections/blacklist
```

The request body is a comma-separated list of the names for the nodes that you want to deny, including any currently denied nodes that should remain denied. Format the request body as an array of string, with each node name in double quotes. Use a comma to separate each node name in the list.

Any nodes omitted from the request body, including currently denied nodes, will be allowed.

Example Request

```
curl -X POST 'https://{hostname}/api/v3/nodeCollections/blacklist' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '["localhost-1.c.dremio-1093.external",
"localhost-2.c.dremio-1093.external"]'
```

Example Response

```
[
  "localhost-1.c.dremio-1093.external",
  "localhost-2.c.dremio-1093.external"
]
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Retrieving Denied Nodes

Invalidate the LDAP authorization for a specific user or group by ID.

Method and URL

```
GET /api/v3/nodeCollections/blacklist
```

Example Request

```
curl -X GET 'https://{hostname}/api/v3/nodeCollections/blacklist' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
[
  "localhost-1.c.dremio-1093.external",
  "localhost-2.c.dremio-1093.external"
]
```


Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Allowing All Nodes

Allow access to all Dremio cluster nodes.

Method and URL

```
POST /api/v3/nodeCollections/blacklist
```

To allow all nodes, send an empty array in the request body.

Example Request

```
curl -X POST 'https://{hostname}/api/v3/nodeCollections/blacklist' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '[]'
```

Example Response

```
[]
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Was this page helpful?

[\[Previous](#)

[LDAP User Cache\]\(/25.x/reference/api/ldap-authorization\)\[Next
Reflection\]\(/25.x/reference/api/reflections/\)](#)

[Denying Nodes](#)

[Response Status Codes](#)

[Retrieving Denied Nodes](#)

[Response Status Codes](#)

Allowing All Nodes

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/reflections/>

Version: 25.x

On this page

Reflection

Use the Reflection API to retrieve a list of raw and aggregation Reflections, retrieve individual Reflections, and create, update, and delete Reflections.

A Reflection is an optimized materialization of source data or a query, similar to a materialized view, that is derived from an existing table or view. The query optimizer can accelerate queries by using one or more Reflections to partially or entirely satisfy the queries rather than running queries against the raw data in the data source that underlies the table or view.

Reflection Object (Raw Reflection)

```
{
  "id": "7a380a24-3b63-436c-9ea0-63cb534cc404",
  "type": "RAW",
  "name": "Raw Reflection",
  "tag": "ureIY76RT7Y=",
  "createdAt": "2023-01-30T14:11:43.826Z",
  "updatedAt": "2023-01-30T14:11:43.826Z",
  "datasetId": "tk973df7-ddf7-4d1e-fa9e-bccf28ae253f",
  "currentSizeBytes": 4393709246,
  "totalSizeBytes": 4393709246,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "failureCount": 0,
    "lastDataFetch": "2023-01-30T14:11:51.801Z",
    "expiresAt": "2023-01-30T17:11:51.801Z"
  },
  "displayFields": [
    {
      "name": "pickup_datetime"
    },
    {
      "name": "passenger_count"
    }
  ],
}
```

```

    {
      "name": "trip_distance_mi"
    },
    {
      "name": "fare_amount"
    },
    {
      "name": "tip_amount"
    },
    {
      "name": "total_amount"
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "partitionFields": [
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "pickup_datetime"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
}

```

Reflection Object (Aggregation Reflection)

```

{
  "id": "95dda9dd-2371-467f-b68d-fc4c5ea57a8b",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "ZpzGgxw2l04=",
  "createdAt": "2022-07-05T19:19:40.244Z",
  "updatedAt": "2023-01-10T17:12:40.244Z",
  "datasetId": "df99ab32-c2d4-4d1c-9e91-2c8be861bb8a",
  "currentSizeBytes": 18639885,
  "totalSizeBytes": 142639924,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",

```

```

    "failureCount": 0,
    "lastDataFetch": "2023-01-10T17:12:40.244Z",
    "expiresAt": "3022-07-05T19:19:40.244Z"
  },
  "dimensionFields": [
    {
      "name": "pickup_date"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "dropoff_date"
    },
    {
      "name": "dropoff_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "surcharge",
      "measureTypeList": [

```

```

        "SUM",
        "COUNT"
    ],
    },
    {
        "name": "tip_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    }
],
"distributionFields": [
    {
        "name": "trip_distance_mi"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "dropoff_date"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Reflection Attributes

id String (UUID)

Unique identifier of the Reflection.

Example: 95dda9dd-2371-467f-b68d-fc4c5ea57a8b

type String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio UI, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: Aggregation Reflection

tag String

Unique identifier of the Reflection instance. Dremio changes the tag whenever the Reflection changes and uses the tag to ensure that PUT requests apply to the most recent version of the Reflection.

Example: ZpzGgxw2l04=

createdAt String

Date and time that the Reflection was created, in UTC format.

Example: 2022-07-05T19:19:40.244Z

updatedAt String

Date and time that the Reflection was last updated, in UTC format.

Example: 2023-01-10T17:12:40.244Z

datasetId String (UUID)

Unique identifier of the anchor dataset that is associated with the Reflection.

Example: df99ab32-c2d4-4d1c-9e91-2c8be861bb8a

currentSizeBytes Integer

Data size of the latest Reflection job (if one exists), in bytes.

Example: 18639885

totalSizeBytes Integer

Data size of all Reflection jobs that have not been pruned (if any exist), in bytes.

Example: 142639924

enabled Boolean

If the Reflection is available for accelerating queries, the value is `true`. Otherwise, the

value is `false`.

Example: true

arrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, the value is `true`. Otherwise, the value is `false`.

Example: false

status Object

Information about the status of the Reflection.

Example: `{ "config": "OK", "refresh": "SCHEDULED", "availability": "AVAILABLE", "combinedStatus": "CAN_ACCELERATE", "failureCount": 0, "lastDataFetch": "2023-01-10T17:12:40.244Z", "expiresAt": "3022-07-05T19:19:40.244Z" }`

displayFields Array of Object

Information about the fields displayed from the anchor dataset. Each displayFields object contains one attribute: name. Valid only for raw Reflections.

Example: `[{ "name": "pickup_datetime" }, { "name": "passenger_count" }, { "name": "trip_distance_mi" }, { "name": "fare_amount" }, { "name": "tip_amount" }, { "name": "total_amount" }]`

dimensionFields Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each dimensionFields object contains two attributes: name and granularity. Valid only for aggregation Reflections.

Example: `[{ "name": "pickup_date", "granularity": "DATE" }, { "name": "pickup_datetime", "granularity": "DATE" }, { "name": "dropoff_date", "granularity": "DATE" }, { "name": "dropoff_datetime", "granularity": "DATE" }, { "name": "passenger_count", "granularity": "DATE" }, { "name": "total_amount", "granularity": "DATE" }]`

measureFields Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each measureFields object contains two attributes: name and measureTypeList. Valid only for aggregation Reflections.

Example: `[{ "name": "passenger_count", "measureTypeList": ["SUM", "COUNT"] }, { "name": "trip_distance_mi", "measureTypeList": ["SUM", "COUNT"] }, { "name": "fare_amount", "measureTypeList": ["SUM", "COUNT"] }, { "name": "surcharge", "measureTypeList": ["SUM", "COUNT"] }, { "name": "tip_amount", "measureTypeList": ["SUM", "COUNT"] }, { "name": "total_amount", "measureTypeList": ["SUM", "COUNT"] }]`

distributionFields Array of Object

Information about the distribution fields from the anchor dataset used in the Reflection. Distribution fields allow data from multiple datasets to be co-located and co-partitioned across nodes to minimize data movement during join operations. Each distributionFields object contains one attribute: name.

Example: [{"name": "trip_distance_mi"}, {"name": "total_amount"}]

partitionFields Array of Object

Information about the fields from the anchor dataset used to partition data in the Reflection. Each field name is listed as an individual object. For more information, read [Horizontally Partition Reflections that Have Many Rows](#).

Example: [{"name": "dropoff_date"}, {"name": "passenger_count"}]

sortFields Array of Object

Information about the fields from the anchor dataset used for sorting in the Reflection. Each sortFields object contains one attribute: name. For more information, read [Sort Reflections on High-Cardinality Fields](#).

Example: [{"name": "trip_distance_mi"}]

partitionDistributionStrategy String

Method used to optimize data compression when executing Reflections. `CONSOLIDATED` means Dremio minimizes the number of files produced. The query threads pool the data and ensure that the fewest number of files are written to the Reflection store. Optimizing for a smaller number of files generally improves read performance because users can perform fewer searches for a given query. `STRIPED` means Dremio minimizes the time required to refresh the Reflection. Each final-stage query thread opens its own writers to write the data, which can result in many small files if each query thread contains a small amount of data.

Enum: CONSOLIDATED, STRIPED

Example: CONSOLIDATED

canView Boolean

If you can view Reflections on all datasets of a source, system, space, or folder, the value is `true`. Otherwise, the value is `false`.

Example: true

canAlter Boolean

If you can create, edit, and view Reflections on all datasets of a source, system, space, or folder, the value is `true`. Otherwise, the value is `false`.

Example: true

entityType String

Type of the object. For Reflection objects, the entityType is `reflection`.

Example: reflection

Attributes of the `status` Object

config String

Status of the Reflection configuration. If the value is `OK`, the Reflection configuration is free of errors. If the value is `INVALID`, the Reflection configuration contains one or more errors.

Enum: OK, INVALID

Example: OK

refresh String

Status of the Reflection refresh.

`GIVEN_UP`: Dremio attempted to refresh the Reflection multiple times, but each attempt has failed and Dremio will not make further attempts.

`MANUAL`: Refresh period is set to 0, so you must use the Dremio UI to manually refresh the Reflection.

`RUNNING`: Dremio is currently refreshing the Reflection.

`SCHEDULED`: The Reflection refreshes according to a schedule.

Enum: GIVEN_UP, MANUAL, RUNNING, SCHEDULED

Example: SCHEDULED

availability String

Status of the Reflection's availability for accelerating queries.

Enum: NONE, INCOMPLETE, EXPIRED, AVAILABLE

Example: AVAILABLE

combinedStatus String

Status of the Reflection based on a combination of config, refresh, and availability.

`CAN_ACCELERATE`: The Reflection is fully functional.

`CAN_ACCELERATE_WITH_FAILURES`: The most recent refresh failed to obtain a status, but Dremio still has a valid materialization.

`CANNOT_ACCELERATE_MANUAL`: The Reflection is unable to accelerate any queries, and the `Never Refresh` option is selected for the refresh policy.

`CANNOT_ACCELERATE_SCHEDULED`: The Reflection is currently unable to accelerate any queries, but it has been scheduled for a refresh at a future time.

`DISABLED`: The Reflection has been manually disabled.

`EXPIRED`: The Reflection has expired and cannot be used.

`FAILED`: The attempt to refresh the Reflection has failed, typically three times in a

row. The Reflection is still usable.

``INVALID``: The Reflection is invalid because the underlying dataset has changed.

``INCOMPLETE``: One or more pseudo-distributed file system (PDFS) nodes that contain materialized files are down (PFDS is supported for v21 and earlier). Only partial data is available. Configurations that use the Hadoop Distributed File System (HDFS) to store Reflections should not experience incomplete status.

``REFRESHING``: The Reflection is currently being refreshed.

Example: `CAN_ACCELERATE`

failureCount Integer

Number of times that an attempt to refresh the Reflection failed.

Example: 0

lastDataFetch String

Date and time that the Reflection data was last refreshed, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetch value is ``1969-12-31T23:59:59.999Z``.

Example: `2023-01-10T17:12:40.244Z`

expiresAt String

Date and time that the Reflection expires, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetch value is ``1969-12-31T23:59:59.999Z``.

Example: `3022-07-05T19:19:40.244Z`

Attributes of Objects in the ``displayFields`` Array

name String

Name of the field from the anchor dataset that is displayed in the raw Reflection.

Example: `passenger_count`

Attributes of Objects in the ``dimensionFields`` Array

name String

Name of the field from the anchor dataset that is configured as a dimension for the Reflection.

Example: `pickup_date`

granularity String

Grouping used for the dimension field. When timestamp and date fields are configured as dimensions, Dremio can automatically extract and use the day-level date value (``DATE``) or use the field's original value (``NORMAL``).

Enum: DATE, NORMAL

Example: DATE

Attributes of Objects in the `measureFields` Array

name String

Name of the field from the anchor dataset that is configured as a measure for the Reflection.

Example: passenger_count

measureTypeList Array of String

Types of calculations for which Dremio uses the specified measure field.

Enum: APPROX_COUNT_DISTINCT, MIN, MAX, UNKNOWN, SUM, COUNT

Example: ["SUM","COUNT"]

Attributes of Objects in the `distributionFields` Array

name String

Name of the field from the anchor dataset that is used for co-locating and co-partitioning data from multiple datasets across nodes.

Example: trip_distance_mi

Attributes of Objects in the `partitionFields` Array

name String

Name of the field from the anchor dataset on which the rows in the Reflection are to be partitioned. If a column is listed as a partition column, it cannot also be listed as a sort column for the same Reflection. In aggregation Reflections, each column specified as a partition column or used in transform must also be listed as a dimension column. In raw Reflections, each column specified as a partition column or used in transform must also be listed as a display column.

Example: trip_distance_mi

transform Object

The type of partition transform that is applied. The value is an enum. The types are:

IDENTITY: Creates one partition per value. This is the default transform. If no transform is specified for a field named by the `name` property, an IDENTITY transform is performed.

IDENTITY Example

```
{  
  "name": "passenger_count",
```

```
"transform": {  
  "type": "IDENTITY"  
}  
}
```

YEAR: Partitions by year. The field must use the TIMESTAMP or DATE data type.

YEAR Example

```
{  
  "name": "pickup_datetime",  
  "transform": {  
    "type": "YEAR"  
  }  
}
```

MONTH: Partitions by month. The field must use the TIMESTAMP or DATE data type.

MONTH Example

```
{  
  "name": "pickup_datetime",  
  "transform": {  
    "type": "MONTH"  
  }  
}
```

DAY: Partitions on the equivalent of dateint. The field must use the TIMESTAMP or DATE data type.

DAY Example

```
{  
  "name": "pickup_datetime",  
  "transform": {  
    "type": "DAY"  
  }  
}
```

HOURL: Partitions on the equivalent of dateint and hour. The field must use the TIMESTAMP data type.

HOURL Example

```
{  
  "name": "pickup_datetime",  
  "transform": {  
    "type": "HOURL"  
  }  
}
```

BUCKET: Partitions data into the number of partitions specified by an integer. For example, if the integer value N is specified, the data is partitioned into N, or (0 to (N-1)), partitions. The partition in which an individual row is stored is determined by hashing the field value and then calculating `<hash_value> mod N`. If the result is 0, the row is placed in partition 0; if the result is 1, the row is placed in partition 1; and so on.

This value must be followed by a ``bucketTransform`` object. This object takes one property: ``bucketCount``. This property takes an integer value.

BUCKET Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "BUCKET",
    "bucketTransform": {
      "bucketCount": 1000
    }
  }
}
```

TRUNCATE: If the specified field uses the string data type, truncates strings to a maximum of the number of characters specified by an integer. For example, suppose the specified transform is ``truncate(1, stateUS)``. A value of ``CA`` is truncated to ``C``, and the row is placed in partition C. A value of ``CO`` is also truncated to ``C``, and the row is also placed in partition C.

If the specified field uses the integer or long data type, truncates field values in the following way: For any ``truncate(L, col)``, truncates the field value to the biggest multiple of L that is smaller than the field value. For example, suppose the specified transform is ``truncate(10, intField)``. A value of 1 is truncated to 0 and the row is placed in the partition 0. A value of 247 is truncated to 240 and the row is placed in partition 240. If the transform is ``truncate(3, intField)``, a value of 13 is truncated to 12 and the row is placed in partition 12. A value of 255 is not truncated, because it is divisible by 3, and the row is placed in partition 255. This value must be followed by a ``truncateTransform`` object.

This object takes one property: ``truncateLength``. This property takes an integer value.

note

The truncate transform does not change field values. It uses field values to calculate the correct partitions in which to place rows.

TRUNCATE Example

```
{
  "name": "pickup_hour",
  "transform": {
    "type": "TRUNCATE",
    "truncateTransform": {
      "truncateLength": 3
    }
  }
}
```

```
}
```

Attributes of Objects in the `sortFields` Array

name String

Name of the field from the anchor dataset that is used for sorting in the Reflection.

Example: dropoff_date

Creating a Reflection

Create a new Reflection.

Method and URL

```
POST /api/v3/reflection
```

Parameters

type Body String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name Body String

Name to use for the Reflection.

Example: New Aggregation Reflection

datasetId Body String (UUID)

Unique identifier of the anchor dataset to associate with the Reflection.

Example: df99ab32-c2d4-4d1c-9e91-2c8be861bb8a

enabled Body Boolean

If the Reflection should be available for accelerating queries, set to `true`. Otherwise, set to `false`.

Example: true

arrowCachingEnabled Body Boolean Optional

If Dremio should convert data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, set to `true`. Otherwise, set to `false` (default).

Example: false

displayFields Body Array of Object Optional

Information about the fields to display from the anchor dataset. The displayfields array must list every field in the anchor dataset or the Reflection fails. Each displayFields object contains one attribute: name. Valid only for raw Reflections.

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}, {"name": "trip_distance_mi"}, {"name": "fare_amount"}, {"name": "tip_amount"}, {"name": "total_amount"}]

dimensionFields Body Array of Object Optional

Information about the dimension fields from the anchor dataset to use in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each dimensionFields object contains two attributes: name and granularity. Valid only for aggregation Reflections.

Example: [{"name": "pickup_datetime", "granularity": "DATE"}, {"name": "passenger_count", "granularity": "DATE"}, {"name": "total_amount", "granularity": "DATE"}, {"name": "trip_distance_mi", "granularity": "DATE"}]

measureFields Body Array of Object Optional

Information about the measure fields from the anchor dataset to use in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each measureFields object contains two attributes: name and measureTypeList. Valid only for aggregation Reflections.

Example: [{"name": "passenger_count", "measureTypeList": ["SUM", "COUNT"]}, {"name": "trip_distance_mi", "measureTypeList": ["SUM", "COUNT"]}, {"name": "fare_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "tip_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "total_amount", "measureTypeList": ["SUM", "COUNT"]}]

distributionFields Body Array of Object Optional

Information about the distribution fields from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. Each distributionFields object contains one attribute: name.

Example: [{"name": "trip_distance_mi"}, {"name": "total_amount"}]

partitionFields Body Array of Object Optional

Information about the fields from the anchor dataset to use to partition data in the Reflection. Each field name is listed as an individual object. For more information, read [Horizontally Partition Reflections that Have Many Rows](#).

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}]

sortFields Body Array of Object Optional

Information about the fields from the anchor dataset to use for sorting in the Reflection. Each sortFields object contains one attribute: name. For more information, read [Sort Reflections on High-Cardinality Fields](#).

Example: [{"name": "trip_distance_mi"}]

partitionDistributionStrategy Body String Optional

Method to use to optimize data compression when executing Reflections. If set to `CONSOLIDATED` (default), Dremio minimizes the number of files produced. If set to `STRIPED`, Dremio minimizes the time required to refresh the Reflection.

Enum: CONSOLIDATED, STRIPED

Example: CONSOLIDATED

canView Body Boolean Optional

To view Reflections on all datasets of a source, system, space, or folder, set to `true` (default). Otherwise, set to `false`.

Example: true

canAlter Body Boolean Optional

To create, edit, and view Reflections on all datasets of a source, system, space, or folder, set to `true` (default). Otherwise, set to `false`.

Example: true

entityType Body String Optional

Type of the object. For Reflection objects, the entityType is `reflection`.

Example: reflection

Parameters of Objects in the `displayFields` Array

name Body String

Name of the field to display from the anchor dataset.

Example: "name": "pickup_datetime"

Parameters of Objects in the `dimensionFields` Array

name Body String

Name of the field from the anchor dataset to configure as a dimension for the Reflection.

Example: "name": "pickup_datetime"

granularity Body String

Grouping to use for the dimension field. If Dremio should automatically extract the day-level date value and use it as the grouping value in the Reflection, `DATE`. If Dremio should use the original value for grouping, `NORMAL`.

Enum: DATE, NORMAL

Example: "granularity": "DATE"

Parameters of Objects in the `measureFields` Array

name Body String

Name of the field from the anchor dataset that you expect to use in calculations. Fields of types `LIST`, `MAP`, and `UNION` are not valid measureFields.

Example: "name": "passenger_count"

measureTypeList Body Array of String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. In aggregation Reflections, every field listed as a distribution field must also be listed as a dimension field.

Enum: APPROX_COUNT_DISTINCT, MIN, MAX, UNKNOWN, SUM, COUNT

Example: ["SUM", "COUNT"]

Parameters of Objects in the `distributionFields` Array

name Body String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. In aggregation Reflections, every field listed as a distribution field must also be listed as a dimension field.

Example: "name": "trip_distance_mi"

Parameters of Objects in the `partitionFields` Array

name Body String

Name of the field from the anchor dataset on which you want to be able to partition rows. If you are creating an aggregation Reflection, every field listed as a partition field must also be listed as a dimension field. If you list a field as a partition field, you cannot list the same field as a sort field in the same Reflection.

Example: "name": "pickup_datetime"

transform Object

The type of partition transform that is applied. The value is an enum. The types are:

IDENTITY: Creates one partition per value. This is the default transform. If no transform is specified for a field named by the `name` property, an IDENTITY transform is performed.

IDENTITY Example

```
{
  "name": "passenger_count",
  "transform": {
```

```
    "type": "IDENTITY"
  }
}
```

YEAR: Partitions by year. The field must use the **TIMESTAMP** or **DATE** data type.

YEAR Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "YEAR"
  }
}
```

MONTH: Partitions by month. The field must use the **TIMESTAMP** or **DATE** data type.

MONTH Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "MONTH"
  }
}
```

DAY: Partitions on the equivalent of `dateint`. The field must use the **TIMESTAMP** or **DATE** data type.

DAY Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "DAY"
  }
}
```

HOURL: Partitions on the equivalent of `dateint` and hour. The field must use the **TIMESTAMP** data type.

HOURL Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "HOURL"
  }
}
```

BUCKET: Partitions data into the number of partitions specified by an integer. For example, if the integer value N is specified, the data is partitioned into N, or (0 to (N-1)), partitions. The partition in which an individual row is stored is determined by hashing the field value and then calculating `<hash_value> mod N`. If the result is 0, the row is placed in partition 0; if the result is 1, the row is placed in partition 1; and so on.

This value must be followed by a ``bucketTransform`` object. This object takes one property: ``bucketCount``. This property takes an integer value.

BUCKET Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "BUCKET",
    "bucketTransform": {
      "bucketCount": 1000
    }
  }
}
```

TRUNCATE: If the specified field uses the string data type, truncates strings to a maximum of the number of characters specified by an integer. For example, suppose the specified transform is ``truncate(1, stateUS)``. A value of ``CA`` is truncated to ``C``, and the row is placed in partition C. A value of ``CO`` is also truncated to ``C``, and the row is also placed in partition C.

If the specified field uses the integer or long data type, truncates field values in the following way: For any ``truncate(L, col)``, truncates the field value to the biggest multiple of L that is smaller than the field value. For example, suppose the specified transform is ``truncate(10, intField)``. A value of 1 is truncated to 0 and the row is placed in the partition 0. A value of 247 is truncated to 240 and the row is placed in partition 240. If the transform is ``truncate(3, intField)``, a value of 13 is truncated to 12 and the row is placed in partition 12. A value of 255 is not truncated, because it is divisible by 3, and the row is placed in partition 255. This value must be followed by a ``truncateTransform`` object.

This object takes one property: ``truncateLength``. This property takes an integer value.

note

The truncate transform does not change field values. It uses field values to calculate the correct partitions in which to place rows.

TRUNCATE Example

```
{
  "name": "pickup_hour",
  "transform": {
    "type": "TRUNCATE",
    "truncateTransform": {
      "truncateLength": 3
    }
  }
}
```

```
}
```

Parameters of Objects in the `sortFields` Array

name Body String

Name of the field from the anchor dataset to use for sorting in the Reflection. Every field listed as a sort field must also be listed as a dimension field. If you list a field as a sort field, you cannot list the same field as a partition field in the same Reflection.

Example: "name": "trip_distance_mi"

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "type": "AGGREGATION",
  "name": "New Aggregation Reflection",
  "datasetId": "gc870df7-ddf7-4d1e-bb9e-beef28ae773f",
  "enabled": true,
  "arrowCachingEnabled": false,
  "dimensionFields": [
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ]
}
```

```

    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    },
    {
      "name": "total_amount"
    }
  ],
  "partitionFields": [
    {
      "name": "pickup_datetime"
    },
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "entityType": "reflection"
}'

```

Example Response

```

{
  "id": "836eae91-306e-487b-a687-31c999653a86",
  "type": "AGGREGATION",
  "name": "New Aggregation Reflection",
  "tag": "sEHieiuiinqE=",

```

```

"createdAt": "2023-01-30T14:30:24.311Z",
"updatedAt": "2023-01-30T14:30:24.311Z",
"datasetId": "gc870df7-ddf7-4d1e-bb9e-beef28ae773f",
"currentSizeBytes": 0,
"totalSizeBytes": 0,
"enabled": true,
"arrowCachingEnabled": false,
"status": {
  "config": "OK",
  "refresh": "SCHEDULED",
  "availability": "NONE",
  "combinedStatus": "CANNOT_ACCELERATE_SCHEDULED",
  "failureCount": 0,
  "lastDataFetch": "1969-12-31T23:59:59.999Z",
  "expiresAt": "1969-12-31T23:59:59.999Z"
},
"dimensionFields": [
  {
    "name": "pickup_datetime",
    "granularity": "DATE"
  },
  {
    "name": "passenger_count"
  },
  {
    "name": "total_amount"
  },
  {
    "name": "trip_distance_mi"
  }
],
"measureFields": [
  {
    "name": "passenger_count",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "trip_distance_mi",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "fare_amount",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  }
]

```

```

        "name": "tip_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    }
],
"distributionFields": [
    {
        "name": "trip_distance_mi"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "pickup_datetime"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Retrieving All Reflections Enterprise

Retrieve a list of Reflection objects that includes all raw and aggregation Reflections in the Dremio instance.

Method and URL

```
GET /api/v3/reflection
```

Example Request

```
curl -X GET 'https://{hostname}/api/v3/reflection/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

In the response for a request to retrieve all raw and aggregation Reflections, the Reflection objects are wrapped with a data array. Each object in the data array represents one Reflection.

Example Response

```
{
  "data": [
    {
      "id": "95dda9dd-2371-467f-b68d-fc4c5ea57a8b",
      "type": "AGGREGATION",
      "name": "Aggregation Reflection",
      "tag": "ZpzGgxw2l04=",
      "createdAt": "2022-07-05T19:19:40.244Z",
      "updatedAt": "2023-01-10T17:12:40.244Z",
      "datasetId": "df99ab32-c2d4-4d1c-9e91-2c8be861bb8a",
      "currentSizeBytes": 18639885,
      "totalSizeBytes": 142639924,
      "enabled": true,
      "arrowCachingEnabled": false,
      "status": {
        "config": "OK",
        "refresh": "SCHEDULED",
        "availability": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "failureCount": 0,
        "lastDataFetch": "2023-01-10T17:12:40.244Z",
        "expiresAt": "3022-07-05T19:19:40.244Z"
      },
      "dimensionFields": [
        {
          "name": "pickup_date"
        },
        {
          "name": "pickup_datetime",
          "granularity": "DATE"
        }
      ]
    }
  ]
}
```



```

        "name": "dropoff_date"
    },
    {
        "name": "dropoff_datetime",
        "granularity": "DATE"
    },
    {
        "name": "passenger_count"
    },
    {
        "name": "total_amount"
    },
    {
        "name": "trip_distance_mi"
    }
],
"measureFields": [
    {
        "name": "passenger_count",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "trip_distance_mi",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "fare_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "surcharge",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "tip_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "total_amount",

```

```

        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    "distributionFields": [
        {
            "name": "trip_distance_mi"
        },
        {
            "name": "total_amount"
        }
    ],
    "partitionFields": [
        {
            "name": "dropoff_date"
        },
        {
            "name": "passenger_count"
        }
    ],
    "sortFields": [
        {
            "name": "trip_distance_mi"
        }
    ],
    "partitionDistributionStrategy": "CONSOLIDATED",
    "canView": true,
    "canAlter": true,
    "entityType": "reflection"
},
{
    "id": "14f22052-cbb3-4d5d-8bbc-6154cca98e49",
    "type": "RAW",
    "name": "listings",
    "tag": "XAY4ccVFX04=",
    "createdAt": "2022-07-12T16:45:35.249Z",
    "updatedAt": "2022-07-12T16:45:35.249Z",
    "datasetId": "7707981c-cb33-42bc-a048-d27a8915f468",
    "currentSizeBytes": 0,
    "totalSizeBytes": 0,
    "enabled": true,
    "arrowCachingEnabled": true,
    "status": {
        "config": "OK",
        "refresh": "MANUAL",
        "availability": "NONE",
        "combinedStatus": "CANNOT_ACCELERATE_MANUAL",
        "failureCount": 0,
        "lastDataFetch": "1969-12-31T23:59:59.999Z",
        "expiresAt": "1969-12-31T23:59:59.999Z"
    },
    "displayFields": [

```

```

    {
      "name": "id"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
},
{
  "id": "6c209200-b522-4f81-bbe0-d10668c7752c",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "SQeEAG3d6DA=",
  "createdAt": "2021-09-29T15:47:44.806Z",
  "updatedAt": "2021-09-29T15:47:44.806Z",
  "datasetId": "746f867a-c27c-4711-bb8c-99546a4c25e0",
  "currentSizeBytes": 0,
  "totalSizeBytes": 1675978,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "GIVEN_UP",
    "availability": "NONE",
    "combinedStatus": "FAILED",
    "failureCount": 3,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "dimensionFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    }
  ],
  "measureFields": [
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],

```

```

    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
},
{
  "id": "c5c5b282-ffea-4a34-835f-cc591584412b",
  "type": "AGGREGATION",
  "name": "Test Reflection",
  "tag": "lMxFcc2qjgE=",
  "createdAt": "2021-10-11T18:44:27.064Z",
  "updatedAt": "2021-10-11T18:44:27.064Z",
  "datasetId": "316531b8-3c56-42f2-b05f-81f228ef3162",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "MANUAL",
    "availability": "NONE",
    "combinedStatus": "CANNOT_ACCELERATE_MANUAL",
    "failureCount": 0,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "dimensionFields": [
    {
      "name": "passenger_count"
    }
  ],
  "measureFields": [
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],

```

```

    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
},
"canAlterReflections": true
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieving a Reflection

Retrieve the specified Reflection.

Method and URL

```
GET /api/v3/reflection/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the Reflection that you want to retrieve.

Example: 95dda9dd-2371-467f-b68d-fc4c5ea57a8b

Example Request

```
curl -X GET 'https://{hostname}/api/v3/reflection/95dda9dd-2371-467f-b68d-fc4c5ea57a8b'

--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "95dda9dd-2371-467f-b68d-fc4c5ea57a8b",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "ZpzGgxw2l04=",
  "createdAt": "2022-07-05T19:19:40.244Z",
  "updatedAt": "2023-01-10T17:12:40.244Z",
  "datasetId": "df99ab32-c2d4-4d1c-9e91-2c8be861bb8a",
  "currentSizeBytes": 18639885,
  "totalSizeBytes": 142639924,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "failureCount": 0,
    "lastDataFetch": "2023-01-10T17:12:40.244Z",
    "expiresAt": "3022-07-05T19:19:40.244Z"
  },
  "dimensionFields": [
    {
      "name": "pickup_date"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "dropoff_date"
    },
    {
      "name": "dropoff_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    }
  ],
}
```

```

    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "surcharge",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    }
  ]
}

```

```

    },
    {
      "name": "total_amount"
    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieving All Reflections for a Dataset

Retrieve all raw and aggregation Reflections for the specified dataset.

Method and URL

```
GET /api/v3/dataset/{datasetId}/reflection
```

Parameters

datasetId Path String (UUID)

Unique identifier of the dataset whose Reflections you want to retrieve.

Example: 3cbab7b3-ee82-44c1-abcc-e86d56078d4d

Example Request

```
curl -X GET 'https://{hostname}/api/v3/dataset/3cbab7b3-ee82-44c1-abcc-e86d56078d4d/reflection' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

In the response for a request to retrieve all Reflections for a dataset, the Reflection objects are wrapped with a data array. Each object in the data array represents one Reflection.

Example Response

```
{
  "data": [
    {
      "id": "23f75eb1-045f-447f-b3fa-374377877569",
      "type": "RAW",
      "name": "Raw Reflection",
      "tag": "K9J2SHE0c+Q=",
      "createdAt": "2023-02-03T16:38:27.770Z",
      "updatedAt": "2023-02-03T16:38:27.770Z",
      "datasetId": "3cbab7b3-ee82-44c1-abcc-e86d56078d4d",
      "currentSizeBytes": 0,
      "totalSizeBytes": 0,
      "enabled": true,
      "arrowCachingEnabled": false,
      "status": {
        "config": "OK",
        "refresh": "MANUAL",
        "availability": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "failureCount": 0,
        "lastDataFetch": "2023-02-03T16:38:27.780Z",
        "expiresAt": "3022-06-06T16:38:27.780Z"
      },
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {
          "name": "fare_amount"
        },
        {
          "name": "tip_amount"
        },
        {
          "name": "total_amount"
        }
      ]
    }
  ]
}
```

```

    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
},
{
  "id": "3cbab7b3-ee82-44c1-abcc-e86d56078d4d",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "Mc4hDFk5JR8=",
  "createdAt": "2023-02-03T16:39:40.556Z",
  "updatedAt": "2023-02-03T16:39:40.556Z",
  "datasetId": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "MANUAL",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "failureCount": 0,
    "lastDataFetch": "2023-02-03T16:39:40.568Z",
    "expiresAt": "3022-06-06T16:39:40.568Z"
  },
  "dimensionFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    }
  ],
  "measureFields": [
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",

```

```

        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    },
    {
        "name": "fare_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}
],
"canAlterReflections": true
}

```

Response Status Codes

- 200 OK
- 401 Unauthorized
- 404 Not Found
- 405 Method Not Allowed
- 500 Internal Server Error

Refreshing a Reflection

For information about the refresh action performed, see [Triggering Refreshes by Using the Reflection API, the Catalog API, or an SQL Command](#).

Method and URL

```
POST /api/v3/reflection/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier for the Reflection that you want to base the refresh action on.

Example: 836eae91-306e-487b-a687-31c999653a86

Response Status Codes

200 OK

400 Not supported

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Updating a Reflection

Update the specified Reflection.

Method and URL

```
PUT /api/v3/reflection/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the Reflection that you want to update.

Example: 836eae91-306e-487b-a687-31c999653a86

type Body String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name Body String

Name to use for the Reflection.

Example: New Aggregation Reflection

tag Body String

Unique identifier of the most recent version of the Reflection. Dremio uses the tag to ensure that you are updating the most recent version of the Reflection.

Example: ZpzGgxw2l04=

datasetId Body String (UUID)

Unique identifier of the anchor dataset associated with the Reflection.

Example: gc870df7-ddf7-4d1e-bb9e-beef28ae773f

enabledBody Boolean

If the Reflection should be available for accelerating queries, set to `true`. Otherwise, set to `false`.

Example: false

arrowCachingEnabledBody Boolean Optional

If Dremio should convert data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, set to `true`. Otherwise, set to `false` (default).

Example: true

displayFields Body Array of Object

Information about the fields to display from the anchor dataset. The displayFields array must list every field in the anchor dataset or the Reflection fails. Each displayFields object contains one attribute: name. Valid only for raw Reflections.

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}, {"name": "trip_distance_mi"}, {"name": "fare_amount"}, {"name": "tip_amount"}, {"name": "total_amount"}]

dimensionFields Body Array of Object

Information about the dimension fields from the anchor dataset to use in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each dimensionFields object contains two attributes: name and granularity. Valid only for aggregation Reflections. If you omit the dimensionFields object in a PUT request, Dremio removes all existing dimension fields from the Reflection. To keep existing dimension fields while making other updates, duplicate the existing dimensionFields array in the PUT request.

Example: [{"name": "pickup_datetime", "granularity": "DATE"}, {"name": "passenger_count", "granularity": "DATE"}, {"name": "total_amount", "granularity": "DATE"}, {"name": "trip_distance_mi", "granularity": "DATE"}]

measureFields Body Array of Object

Information about the measure fields from the anchor dataset to use in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each measureFields object contains two attributes: name and measureTypeList. Valid only for aggregation Reflections. If you omit the measureFields object in a PUT request, Dremio removes all existing measure fields from the Reflection. To keep existing measure fields while making other updates, duplicate the existing measureFields array in the PUT request.

Example: [{"name": "passenger_count", "measureTypeList": ["SUM", "COUNT"]}, {"name": "trip_distance_mi", "measureTypeList": ["SUM", "COUNT"]}, {"name": "fare_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "tip_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "total_amount", "measureTypeList": ["SUM", "COUNT"]}]

distributionFields Body Array of Object Optional

Information about the distribution fields from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. Each distributionFields object contains one attribute: name.

If you omit the distributionFields object in a PUT request, Dremio removes all existing distribution fields from the Reflection. To keep existing distribution fields while making other updates, duplicate the existing distributionFields array in the PUT request.

Example: [{"name": "trip_distance_mi"}, {"name": "total_amount"}]

partitionFields Body Array of Object Optional

Information about the fields from the anchor dataset to use to partition data in the Reflection. Each field name is listed as an individual object. If you omit the partitionFields object in a PUT request, Dremio removes all existing partition fields from the Reflection. To keep existing partition fields while making other updates, duplicate the existing partitionFields array in the PUT request. For more information, read [Horizontally Partition Reflections that Have Many Rows](#).

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}]

sortFields Body Array of Object

Information about the fields from the anchor dataset to use for sorting in the Reflection. Each sortFields object contains one attribute: name. If you omit the sortFields object in a PUT request, Dremio removes all existing sort fields from the Reflection. To keep existing sort fields while making other updates, duplicate the existing sortFields array in the PUT request. For more information, read [Sort Reflections on High-Cardinality Fields](#).

Example: "name": "trip_distance_mi"

partitionDistributionStrategy Body String Optional

Method to use to optimize data compression when executing Reflections. If set to `CONSOLIDATED` (default), Dremio minimizes the number of files produced. If set to `STRIPED`, Dremio minimizes the time required to refresh the Reflection.

Enum: CONSOLIDATED, STRIPED

Example: CONSOLIDATED

Parameters of Objects in the `displayFields` Array

name Body String

Name of the field to display from the anchor dataset.

Example: "name": "pickup_datetime"

Parameters of Objects in the `dimensionFields` Array

name Body String

Name of the field from the anchor dataset to configure as a dimension for the

Reflection.

Example: "name": "pickup_datetime"

granularity Body String

Grouping to use for the dimension field. If Dremio should automatically extract the day-level date value and use it as the grouping value in the Reflection, `DATE`. If Dremio should use the original value for grouping, `NORMAL`.

Enum: DATE, NORMAL

Example: "granularity": "DATE"

Parameters of Objects in the `measureFields` Array

name Body String

Name of the field from the anchor dataset that you expect to use in calculations. Fields of types `LIST`, `MAP`, and `UNION` are not valid measureFields.

Example: "name": "passenger_count"

measureTypeList Body Array of String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. Every field listed as a distribution field must also be listed as a dimension field.

Enum: APPROX_COUNT_DISTINCT, MIN, MAX, UNKNOWN, SUM, COUNT

Example: ["SUM", "COUNT"]

Parameters of Objects in the `distributionFields` Array

name Body String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. In aggregation Reflections, every field listed as a distribution field must also be listed as a dimension field.

Example: "name": "pickup_datetime"

Parameters of Objects in the `partitionFields` Array

name Body String

Name of the field from the anchor dataset on which the rows in the Reflection are to be partitioned. If a column is listed as a partition column, it cannot also be listed as a sort column for the same Reflection. In aggregation Reflections, each column specified as a partition column or used in transform must also be listed as a dimension column. In raw Reflections, each column specified as a partition column or used in transform must also be listed as a display column.

Example: "name": "dropoff_date"

transform Object

The type of partition transform that is applied. The value is an enum. The types are:

IDENTITY: Creates one partition per value. This is the default transform. If no transform is specified for a field named by the `name` property, an IDENTITY transform is performed.

IDENTITY Example

```
{
  "name": "passenger_count",
  "transform": {
    "type": "IDENTITY"
  }
}
```

YEAR: Partitions by year. The field must use the TIMESTAMP or DATE data type.

YEAR Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "YEAR"
  }
}
```

MONTH: Partitions by month. The field must use the TIMESTAMP or DATE data type.

MONTH Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "MONTH"
  }
}
```

DAY: Partitions on the equivalent of dateint. The field must use the TIMESTAMP or DATE data type.

DAY Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "DAY"
  }
}
```

HOURL: Partitions on the equivalent of dateint and hour. The field must use the

TIMESTAMP data type.

HOURL Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "HOURL"
  }
}
```

BUCKET: Partitions data into the number of partitions specified by an integer. For example, if the integer value N is specified, the data is partitioned into N, or (0 to (N-1)), partitions. The partition in which an individual row is stored is determined by hashing the field value and then calculating `<hash_value> mod N`. If the result is 0, the row is placed in partition 0; if the result is 1, the row is placed in partition 1; and so on.

This value must be followed by a ``bucketTransform`` object. This object takes one property: ``bucketCount``. This property takes an integer value.

BUCKET Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "BUCKET",
    "bucketTransform": {
      "bucketCount": 1000
    }
  }
}
```

TRUNCATE: If the specified field uses the string data type, truncates strings to a maximum of the number of characters specified by an integer. For example, suppose the specified transform is ``truncate(1, stateUS)``. A value of ``CA`` is truncated to ``C``, and the row is placed in partition C. A value of ``CO`` is also truncated to ``C``, and the row is also placed in partition C.

If the specified field uses the integer or long data type, truncates field values in the following way: For any ``truncate(L, col)``, truncates the field value to the biggest multiple of L that is smaller than the field value. For example, suppose the specified transform is ``truncate(10, intField)``. A value of 1 is truncated to 0 and the row is placed in the partition 0. A value of 247 is truncated to 240 and the row is placed in partition 240. If the transform is ``truncate(3, intField)``, a value of 13 is truncated to 12 and the row is placed in partition 12. A value of 255 is not truncated, because it is divisible by 3, and the row is placed in partition 255. This value must be followed by a ``truncateTransform`` object.

This object takes one property: ``truncateLength``. This property takes an integer value.

note

The truncate transform does not change field values. It uses field values to calculate

the correct partitions in which to place rows.

TRUNCATE Example

```
{
  "name": "pickup_hour",
  "transform": {
    "type": "TRUNCATE",
    "truncateTransform": {
      "truncateLength": 3
    }
  }
}
```

Parameters of Objects in the `sortFields` Array

name Body String

Name of the field from the anchor dataset to use for sorting in the Reflection. Every field listed as a sort field must also be listed as a dimension field. If you list a field as a sort field, you cannot list the same field as a partition field in the same Reflection.

Example: "name": "pickup_datetime"

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/reflection/836eae91-306e-487b-a687-31c999653a86' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "id": "836eae91-306e-487b-a687-31c999653a86",
  "type": "AGGREGATION",
  "name": "New Aggregation Reflection",
  "tag": "sEHieiuinqE=",
  "datasetId": "gc870df7-ddf7-4dle-bb9e-beef28ae773f",
  "enabled": false,
  "arrowCachingEnabled": true,
  "dimensionFields": [
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
```

```

    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    },
    {
      "name": "total_amount"
    }
  ],
  "partitionFields": [
    {
      "name": "pickup_datetime"
    },
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {

```

```

      "name": "trip_distance_mi"
    }
  ],
  "entityType": "reflection"
}'

```

Example Response

```

{
  "id": "836eae91-306e-487b-a687-31c999653a86",
  "type": "AGGREGATION",
  "name": "New Aggregation Reflection",
  "tag": "nRPbilwodqC=",
  "createdAt": "2023-01-30T14:35:19.192Z",
  "updatedAt": "2023-01-30T14:35:19.192Z",
  "datasetId": "gc870df7-ddf7-4d1e-bb9e-beef28ae773f",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": false,
  "arrowCachingEnabled": true,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "NONE",
    "combinedStatus": "DISABLED",
    "failureCount": 0,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "dimensionFields": [
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {

```

```

        "name": "trip_distance_mi",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "fare_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "tip_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    }
],
"distributionFields": [
    {
        "name": "trip_distance_mi"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "pickup_datetime"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Deleting a Reflection

Delete the specified Reflection.

Method and URL

```
DELETE /api/v3/reflection/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the Reflection that you want to delete.

Example: 95dda9dd-2371-467f-b68d-fc4c5ea57a8b

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/reflection/95dda9dd-2371-467f-b68d-fc4c5ea57a8b' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type:application/json'
```

Example Response

```
No response
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Node Collections](/25.x/reference/api/nodeCollections/)[Next

Recommendations](/25.x/reference/api/reflections/reflection-recommendations)

Reflection Attributes

Creating a Reflection

Parameters

Response Status Codes

Retrieving All Reflections Enterprise

Response Status Codes

Retrieving a Reflection

Parameters

Response Status Codes

Retrieving All Reflections for a Dataset

Parameters

Response Status Codes

Refreshing a Reflection

Parameters

Response Status Codes

Updating a Reflection

Parameters

Response Status Codes

Deleting a Reflection

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/role>
s/

Version: 25.x

On this page

Role Enterprise

Use the Role API to manage roles.

Role Object

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
      "name": "qa_team1",
      "type": "INTERNAL"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 3,
  "description": "Role for testing the new feature"
}
```

Role Attributes

id String (UUID)

Unique identifier of the role.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

name String

User-provided name of the role.

Example: Temporary Testing

type String

Origin of the role.

~INTERNAL~: Role was created in the Dremio user interface (UI) or with the Role API.

~EXTERNAL~: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

~SYSTEM~: Role was predefined in Dremio.

Example: INTERNAL

roles Array of Object

Information about the roles to which the role belongs.

Example: [{"id": "6f87a9c5-d733-4935-8331-875a4a8e09d7","name": "SELECT and CREATE","type": "INTERNAL"}, {"id": "f8426061-8413-46ec-a84d-1b481a97b248","name": "VIEW","type": "INTERNAL"}]

memberCount Integer

Number of users and roles that are members of the role.

Example: 3

description String

User-provided description of the role.

Example: Role for testing the new feature

Attributes of Objects in the `roles` Array

id String (UUID)

Unique identifier of the role.

Example: 6f87a9c5-d733-4935-8331-875a4a8e09d7

name String

Name of the role.

Example: SELECT and CREATE

type String

Origin of the role.

`INTERNAL`: Role was created in the Dremio user interface (UI) or with the Role API.

`EXTERNAL`: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

`SYSTEM`: Role was predefined in Dremio.

Example: INTERNAL

Creating a Role

Create a Dremio role.

Method and URL

POST /api/v3/role

Parameters

name Body String

Name for the role. The role name must be unique and cannot be updated after the role is created.

Example: Temporary Testing

roles Body Array of Object Optional

Information about the roles to which the role should be assigned.

Example: [{"id": "6f87a9c5-d733-4935-8331-875a4a8e09d7"}, {"id": "f8426061-8413-46ec-a84d-1b481a97b248"}]

description Body String Optional

Description for the role.

Example: Role for testing the new feature

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role to which the role you create should be assigned.

Example: 6f87a9c5-d733-4935-8331-875a4a8e09d7

name Body String Optional

Name of the role to which the role you create should be assigned.

Example: qa_team1

Example Request

```
curl -X POST 'https://{hostname}/api/v3/role' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "Temporary Testing",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248"
    }
  ],
  "description": "Role for testing the new feature"
}'
```

Example Response

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
```

```

"name": "Temporary Testing",
"type": "INTERNAL",
"roles": [
  {
    "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
    "name": "qa_team1",
    "type": "INTERNAL"
  },
  {
    "id": "f8426061-8413-46ec-a84d-1b481a97b248",
    "name": "prod_testing",
    "type": "INTERNAL"
  }
],
"memberCount": 0,
"description": "Role for testing the new feature"
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

405 Method Not Allowed

Retrieving a Role by ID

Retrieve a specific role by the role's ID.

Method and URL

```
GET /api/v3/role/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the role you want to retrieve.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

Example Request

```

curl -X GET 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'

```

Example Response

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
      "name": "qa_team1",
      "type": "INTERNAL"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 3,
  "description": "Role for testing the new feature"
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieving a Role by Name

Retrieve a specific role by the role's name.

Method and URL

```
GET /api/v3/role/by-name/{name}
```

Parameters

name Path String

Name of the role you want to retrieve. The role name is case-insensitive. If the role name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Temporary%20Testing

Example Request

```
curl -X GET 'https://{hostname}/api/v3/role/by-name/Temporary%20Testing' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{  
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",  
  "name": "Temporary Testing",  
  "type": "INTERNAL",  
  "roles": [  
    {  
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",  
      "name": "qa_team1",  
      "type": "INTERNAL"  
    },  
    {  
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",  
      "name": "prod_testing",  
      "type": "INTERNAL"  
    }  
  ],  
  "memberCount": 3,  
  "description": "Role for testing the new feature"  
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Updating a Role

Update the specified role.

Method and URL

```
PUT /api/v3/role/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the role you want to update.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

id Body String (UUID)

Unique identifier of the role you want to update.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

name Body String

Name of the role.

Example: Temporary Testing

roles Body Array of Object Optional

Information about the roles to which the role should be assigned. If you omit an existing role in a PUT request, Dremio removes the role. To keep all existing roles while making other updates, include all existing roles in the PUT request.

Example: [{"id": "f8426061-8413-46ec-a84d-1b481a97b248"}]

description Body String Optional

Description to use for the role. If you omit the description in a PUT request, Dremio removes the existing description. To keep the existing description while making other updates, include the description in the PUT request.

Example: Role for viewing the new feature

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role to which the role you update should be assigned.

Example: f8426061-8413-46ec-a84d-1b481a97b248

name Body String Optional

Name of the role to which the role you update should be assigned.

Example: prod_testing

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "roles": [
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248"
    }
  ]
}
```

```
],  
  "description": "Role for viewing the new feature"  
}'
```

Example Response

```
{  
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",  
  "name": "Temporary Testing",  
  "type": "INTERNAL",  
  "roles": [  
    {  
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",  
      "name": "prod_testing",  
      "type": "INTERNAL"  
    }  
  ],  
  "memberCount": 3,  
  "description": "Role for viewing the new feature"  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

405 Method Not Allowed

Add and Remove Role Members

Add and remove members (roles and users) of the specified role.

Method and URL

```
PATCH /api/v3/role/{id}/member
```

Parameters

id Path String (UUID)

Unique identifier of the role for which you want to add or remove members.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

op Body String

Action to take for the user or role.

Enum: add, remove

Example: add

id Body String (UUID)

Unique identifier of the user or role to add or remove.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

type Body String

Type of member you want to add or remove.

Enum: role, user

Example: role

The request body is an array of objects. Each object includes the three parameters for a single user or role that you want to add or remove:

Example Request

```
curl -X PATCH 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d/member' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '[
{
  "op": "add",
  "id": "f8426061-8413-46ec-a84d-1b481a97b248",
  "type": "role"
},
{
  "op": "add",
  "id": "671cdeb8-1af9-45b6-98ee-8ca1e0543a38",
  "type": "user"
},
{
  "op": "remove",
  "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
  "type": "role"
},
{
  "op": "remove",
  "id": "614a6938-7a69-4f7c-ab96-00b50addb1f9",
  "type": "user"
}
]'
```

Example Response

No response

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

405 Method Not Allowed

Deleting a Role

Delete the specified role.

Method and URL

```
DELETE /api/v3/role/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the role that you want to delete.

note

It is not possible to delete a system role, like ADMIN or PUBLIC. Requests to delete a system role result in a `404 Not Found` response.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Reflection Summary](/25.x/reference/api/reflections/reflection-summary)[Next

Role Privileges](/25.x/reference/api/roles/privilege)

Role Attributes

Creating a Role

Parameters

Response Status Codes

Retrieving a Role by ID

Parameters

Response Status Codes

Retrieving a Role by Name

Parameters

Response Status Codes

Updating a Role

Parameters

Response Status Codes

Add and Remove Role Members

Parameters

Response Status Codes

Deleting a Role

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/scripts/>

Version: 25.x

On this page

Scripts

Use the Scripts API to manage scripts, retrieve the contents of scripts for use in a scheduler, retrieve and update privileges on scripts, and migrate scripts between different environments.

Scripts Object

```
{
  "total": 3,
  "data": [
    {
      "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
      "name": "newScript",
      "content": "SELECT * FROM sampled",
      "context": [
        "@dremio",
        "scriptsFolder"
      ],
      "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "createdAt": "2024-05-24T17:42:00.304Z",
      "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
      "modifiedAt": "2024-05-24T17:42:00.304Z",
      "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
    },
    {
      "id": "37dab994-3f1f-4de7-b2e7-49cb5ff0b395",
      "name": "tmp_testing_04-15-24",
      "content": "-- create table accounting_storage.\"tmp.dremio.com\".test1(id INT);\n-- refresh dataset test1\nalter table test1 REFRESH METADATA;",
      "context": [
        "accounting_storage",
        "tmp.dremio.com"
      ],
      "owner": "ba92bf87-174d-422e-beeb-d526757c8099",
      "createdAt": "2024-04-15T10:38:31.433Z",
      "createdBy": "ba92bf87-174d-422e-beeb-d526757c8099",
      "modifiedAt": "2024-04-15T11:03:27.542Z",
      "modifiedBy": "ba92bf87-174d-422e-beeb-d526757c8099"
    },
    {
      "id": "02fef13e-cedd-46ac-b5bf-abcdcd092146",
      "name": "pop10000_A",
      "content": "SELECT * FROM Samples.\"samples.dremio.com\".\"zips.json\" WHERE pop > 10000 AND STARTS_WITH(city, 'A');\n\nCREATE OR REPLACE VIEW myView AS\nSELECT city, state, pop FROM Samples.\"samples.dremio.com\".\"zips.json\"\nWHERE pop > 10000;\n\nCREATE OR REPLACE VIEW myView2 AS\nSELECT * FROM myView\nWHERE STARTS_WITH(city, 'A');\n\nALTER TABLE myView2\nCREATE RAW Reflection myReflection\nUSING DISPLAY(city, state, pop);\n\nSELECT * FROM myView2;\n\nCREATE OR REPLACE FUNCTION isMA(state VARCHAR)\nRETURNS BOOLEAN\nRETURN SELECT state = 'MA';\n\nALTER TABLE myView ADD ROW ACCESS POLICY isMA(\"state\");\n\nSELECT * FROM myView2;\n\n// Refresh the reflection\n\nSELECT * FROM myView2;",
      "context": [
        "@dev"
      ],
      "owner": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
      "createdAt": "2024-05-16T18:08:06.363Z",
    }
  ]
}
```

```

    "createdBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
    "modifiedAt": "2024-05-16T18:31:22.593Z",
    "modifiedBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921"
  }
]
}

```

Scripts Attributes

total Integer

Total number of scripts in the organization.

Example: 3

data Array of Object

List of the scripts in the organization, with an individual object representing each script.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the script. Generated by Dremio and immutable.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

name String

User-provided name of the script.

Example: newScript

content String

The script's SQL.

Example: SELECT * FROM sampled

context Array of String

Path where the SQL query specified in the content attribute runs. If no context is specified for the script, the value is an empty array.

Example: ["@dremio","scriptsFolder"]

owner String

Unique identifier for the user who owns the script.

Example: 8be516f3-04c4-4d19-824d-5a70b3c4442e

createdAt String

Date and time that the script was created. In UTC format.

Example: 2024-05-24T17:42:00.304Z

createdBy String

The unique identifier for the user who created the script.

Example: a97c694f-1e55-4f34-91aa-97f99fee802e

modifiedAt String

Date and time that the script was last modified. In UTC format.

Example: 2024-05-24T17:42:00.304Z

modifiedBy String

The unique identifier for the user who last modified the script.

Example: a97c694f-1e55-4f34-91aa-97f99fee802e

Creating a Script

Create a script.

Method and URL

```
POST /api/v3/scripts
```

Parameters

name Body String

Name to use for the script.

Example: newScript

content Body String

The SQL for the script.

Example: SELECT * FROM sampledb

context Body Array of String Optional

Path where the SQL query specified in the content attribute should run.

Example: ["@dremio","scriptsFolder"]

owner Body String Optional

Unique identifier for the user who should own the script. Default is the ID for the user who creates the script.

Example: 8be516f3-04c4-4d19-824d-5a70b3c4442e

Example Request

```
curl -X POST 'https://{hostname}/api/v3/scripts' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "newScript",
  "content": "SELECT * FROM sampled",
  "context": [
    "@dremio",
    "scriptsFolder"
  ],
  "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e"
}'
```

Example Response

```
{
  "id": "f873a72e-12a5-4537-a393-f9675da7c5f8",
  "name": "newScript",
  "content": "SELECT * FROM sampled",
  "context": [
    "@dremio",
    "scriptsFolder"
  ],
  "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
  "createdAt": "2024-05-24T17:42:00.304Z",
  "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "modifiedAt": "2024-05-24T17:42:00.304Z",
  "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

Retrieving All Scripts

Retrieve a list of all scripts in the organization.

Method and URL

GET /api/v3/scripts/

Parameters

createdBy Query String Optional

Unique identifier for the user who created the scripts you want to retrieve. For more information, read [createdBy Query Parameter](#).

maxResults Query Integer Optional

Maximum number of scripts to return in the response. Maximum valid value is `100`. Default is `25`. For more information, read [maxResults Query Parameter](#).

offset Query Integer Optional

Number of rows to skip for pagination. Default is `0`. Read [limit and offset Query Parameters](#) for usage examples.

orderBy Query String Optional

Organize the response in ascending (default) or descending order by name, createdAt, or modifiedAt. To specify descending order, precede the orderBy value with a `-` character. To organize the response by more than one attribute, use a comma-separated list: `?orderBy=createdAt,name`. For more information, read [orderBy Query Parameter](#).

ownedBy Query String Optional

Unique identifier for the user who owns the scripts you want to retrieve. For more information, read [ownedBy Query Parameter](#).

search Query String Optional

The string for which to search the values of the name attributes in scripts. Read [search Query Parameter](#) for usage examples.

Example Request

```
curl -X GET 'https://{hostname}/api/v3/scripts' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "total": 3,
  "data": [
    {
      "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
      "name": "newScript",
      "content": "SELECT * FROM sampled",
      "context": [
        "@dremio",
        "scriptsFolder"
      ],
      "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "createdAt": "2024-05-24T17:42:00.304Z",
      "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
      "modifiedAt": "2024-05-24T17:42:00.304Z",
      "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
    },
  ],
}
```

```

{
  "id": "37dab994-3f1f-4de7-b2e7-49cb5ff0b395",
  "name": "tmp_testing_04-15-24",
  "content": "-- create table accounting_storage.\"tmp.dremio.com\".test1(id
INT);\n-- refresh dataset test1\nalter table test1 REFRESH METADATA;",
  "context": [
    "accounting_storage",
    "tmp.dremio.com"
  ],
  "owner": "ba92bf87-174d-422e-becb-d526757c8099",
  "createdAt": "2024-04-15T10:38:31.433Z",
  "createdBy": "ba92bf87-174d-422e-becb-d526757c8099",
  "modifiedAt": "2024-04-15T11:03:27.542Z",
  "modifiedBy": "ba92bf87-174d-422e-becb-d526757c8099"
},
{
  "id": "02fef13e-cedd-46ac-b5bf-abcdcd092146",
  "name": "pop10000_A",
  "content": "SELECT * FROM Samples.\"samples.dremio.com\".\"zips.json\" WHERE pop >
10000 AND STARTS_WITH(city, 'A');\n\nCREATE OR REPLACE VIEW myView AS\nSELECT city,
state,  pop  FROM  Samples.\"samples.dremio.com\".\"zips.json\"\nWHERE  pop  >
10000;\n\nCREATE OR REPLACE VIEW myView2 AS\nSELECT * FROM myView\nWHERE
STARTS_WITH(city, 'A');\n\nALTER TABLE myView2 \nCREATE RAW Reflection myReflection
\nUSING DISPLAY(city, state, pop);\n\nSELECT * FROM myView2;\n\nCREATE OR REPLACE
FUNCTION isMA(state VARCHAR)\n      RETURNS BOOLEAN\n      RETURN SELECT state =
'MA';\n\nALTER TABLE myView ADD ROW ACCESS POLICY isMA(\"state\");\n\nSELECT * FROM
myView2;\n\n// Refresh the reflection\n\nSELECT * FROM myView2;",
  "context": [
    "@dev"
  ],
  "owner": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
  "createdAt": "2024-05-16T18:08:06.363Z",
  "createdBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
  "modifiedAt": "2024-05-16T18:31:22.593Z",
  "modifiedBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921"
}
]
}

```

Response Status Codes

200 OK

401 Unauthorized

Retrieving a Script by ID

Retrieve the specified script.

Method and URL

GET /api/v3/scripts/{id}

Parameters

id Path String (UUID)

Unique identifier of the script you want to retrieve.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

Example Request

```
curl -X GET 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
  "name": "newScript",
  "content": "SELECT * FROM sampled",
  "context": [
    "@dremio",
    "scriptsFolder"
  ],
  "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
  "createdAt": "2024-05-24T17:42:00.304Z",
  "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "modifiedAt": "2024-05-24T17:42:00.304Z",
  "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Updating a Script

Update the specified script.

Method and URL

```
PATCH /api/v3/scripts/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the script.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

name Body String Optional

User-provided name to use for the script.

Example: updatedScript

content Body Array of String Optional

The updated SQL for the script.

Example: SELECT * FROM Samples."samples.dremio.com"."zips.json"

context Body String Optional

Path where the SQL query specified in the content attribute should run.

Example: ["@dremio","secondScriptsFolder"]

owner Body String Optional

Unique identifier for the user who should own the script.

Example: a97c694f-1e55-4f34-91aa-97f99fee802e

Example Request

```
curl -X PATCH 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "updatedScript",
  "content": "SELECT * FROM Samples."samples.dremio.com"."zips.json",
  "context": [
    "@dremio",
    "secondScriptsFolder"
  ],
  "owner": "a97c694f-1e55-4f34-91aa-97f99fee802e"
}'
```

Example Response

```
{
  "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
  "name": "updatedScript",
  "content": "SELECT * FROM Samples.\"samples.dremio.com\".\"zips.json\"",
  "context": [
    "@dremio",
    "secondScriptsFolder"
  ],
  "owner": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "createdAt": "2024-05-24T17:42:00.304Z",
  "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
}
```

```
"modifiedAt": "2024-05-24T18:56:59.409Z",  
"modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

Deleting a Script

Delete the specified script.

Method and URL

```
DELETE /api/v3/scripts/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the script that you want to delete.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e'  
\  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Deleting a Group of Scripts

Delete the listed group of scripts.

Method and URL

```
POST /api/v3/scripts:batchDelete
```

Parameters

ids Body Array of String

Array of unique identifiers of the scripts that you want to delete.

Example:

```
["74cfddfd-cb0b-4b2f-b555-cb8b827fec1e","37dab994-3f1f-4de7-b2e7-49cb5ff0b395","02fef13e-cedd-46ac-b5bf-abcdcd092146"]
```

Example Request

```
curl -X POST 'https://{hostname}/api/v3/scripts:batchDelete' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "ids": [
    "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
    "37dab994-3f1f-4de7-b2e7-49cb5ff0b395",
    "02fef13e-cedd-46ac-b5bf-abcdcd092146"
  ]
}'
```

Example Response

```
{
  "unauthorizedIds": [],
  "notFoundIds": [],
  "otherErrorIds": []
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Retrieving Privilege Information for a Script

Retrieve information about the privileges granted on the specified script.

Method and URL

GET /api/v3/scripts/{id}/grants

Parameters

id Path String (UUID)

Unique identifier of the script whose privilege information you want to retrieve.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

Example Request

```
curl -X GET 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e/grants' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "users": [
    {
      "granteeId": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE",
        "MANAGE_GRANTS"
      ]
    },
    {
      "granteeId": "a97c694f-1e55-4f34-91aa-97f99fee802e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": []
}
```

Response Status Codes

200 OK

401 Unauthorized

Updating Privileges Granted on a Script

Update the privileges that are granted on the specified script.

Method and URL

```
PUT /api/v3/scripts/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the script whose privilege grants you want to update.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

users Body Array of Object Optional

Array of objects that specify which users should have privileges on the script, as well as each user's specific privileges.

roles Body Array of Object Optional

Array of objects that specify which roles should have privileges on the script, as well as each role's specific privileges.

Attributes of Objects in the `users` Array

granteeld Body String Optional

Unique identifier for the user for whom you want to add or update privileges.

Example: 8be516f3-04c4-4d19-824d-5a70b3c4442e

privileges Body Array of String Optional

The array of privileges you want to add or update for the user.

Enum: VIEW, MODIFY, DELETE, MANAGE_GRANTS

Example: ["VIEW","MODIFY","DELETE"]

Attributes of Objects in the `roles` Array

granteeld Body String Optional

Unique identifier for the role for which you want to add or update privileges.

Example: 6a1725a3-5721-44e3-b64f-0b39a35749ab

privileges Body Array of String Optional

The array of privileges you want to add or update for the role.

Enum: VIEW, MODIFY, DELETE, MANAGE_GRANTS

Example: ["VIEW","MODIFY","DELETE"]

Example Request

```
curl -X PUT \
'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e/grants' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "users": [
    {
      "granteeId": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE"
      ]
    }
  ],
  "roles": [
    {
      "granteeId": "6a1725a3-5721-44e3-b64f-0b39a35749ab",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE"
      ]
    }
  ]
}'
```

Example Response

```
{
  "users": [
    {
      "granteeId": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE"
      ]
    }
  ],
  {
    "granteeId": "a97c694f-1e55-4f34-91aa-97f99fee802e",
    "privileges": [
      "VIEW",
      "MODIFY",
      "DELETE",
      "MANAGE\_GRANTS"
    ]
  }
}
```

```

    ]
  }
],
"roles": [
  {
    "granteeId": "6a1725a3-5721-44e3-b64f-0b39a35749ab",
    "privileges": [
      "VIEW",
      "MODIFY",
      "DELETE"
    ]
  }
]
}
]
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Was this page helpful?

[\[Previous](#)

[Role Privileges\]\(/25.x/reference/api/roles/privilege\)\[Next](#)

[Source\]\(/25.x/reference/api/source\)](#)

[Scripts Attributes](#)

[Creating a Script](#)

[Parameters](#)

[Response Status Codes](#)

[Retrieving All Scripts](#)

[Parameters](#)

[Response Status Codes](#)

[Retrieving a Script by ID](#)

[Parameters](#)

[Response Status Codes](#)

[Updating a Script](#)

[Parameters](#)

-

Response Status Codes

Deleting a Script

Parameters

Response Status Codes

Deleting a Group of Scripts

Parameters

Response Status Codes

Retrieving Privilege Information for a Script

Parameters

Response Status Codes

Updating Privileges Granted on a Script

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/source>

Version: 25.x

On this page

Source

Use the Source API to clear the AWS Lake Formation permission cache for AWS Glue Data Catalog sources.

Dremio keeps a cache of permissions defined in AWS Lake Formation with a one-hour expiry time. When the cache for the queried table expires, Dremio requests permission information from AWS Lake Formation. After changing permissions on the AWS Lake Formation side, use the Source API to immediately invalidate Dremio's AWS Lake Formation permission cache.

note

The Source API is supported only for AWS Glue Data Catalog sources.

Clearing the Permission Cache

Clear the AWS Lake Formation permission cache for an AWS Glue Data Catalog source.

Method and URL

```
DELETE /api/v3/source/{source-name}/permission-cache
```

Parameters

source-name Path String

The name of the AWS Glue Data Catalog source whose Lake Formation permission cache you want to clear.

Example: glueProd

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/source/glueProd/permission-cache' \  
--header 'Authorization: Bearer <personal access token>' \  
--header 'Content-Type: application/json'
```

Example Response

No response

Responses

204 No Content

400 Bad Request

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Was this page helpful?

[Previous

Scripts](/25.x/reference/api/scripts/)[Next

SQL](/25.x/reference/api/sql/)

Clearing the Permission Cache

Parameters

Responses

Source:
<https://docs.dremio.com/25.x/reference/api/sql/>

Version: 25.x

On this page

SQL

Use the SQL API to submit SQL queries. The response contains the ID for the job associated with the SQL query. Use the job ID in [Job API](#) requests to get more information about the job, including results.

Submitting an SQL Query

Submit an SQL query and retrieve the associated job ID for use in [Job API](#) requests.

Method and URL

```
POST /api/v3/sql
```

Parameters

sql Body String

SQL query to run.

note

Double-quotation marks within a SQL statement need to be escaped.

Example: `SELECT * FROM Samples."samples.dremio.com"."SF weather 2018-2019.csv"`

context Body Array of String Optional

Path to the container where the query should run within Dremio, expressed as an array. The path consists of the source or space, followed by the folder and subfolders.

Example: `["Samples","samples.dremio.com"]`

[references](#) Body Object Optional

References to the specific versions (branches, tags, and commits) in Nessie sources where you want to run the SQL query. If references are not specified for a Nessie source, the SQL query runs on the default branch.

Example: `{"nessieSource1": {"type": "BRANCH","value": "testing"},"nessieSource2": {"type": "TAG","value": "Test commit"},"nessieSource3": {"type": "COMMIT","value": "7a5edb57e035f52beccfab632cea070514eb8b773f616aaeaf668e2f0be8f10d"}}`

Parameters of the `references` Object

[<Nessie source>](#) Body String Optional

The name of the Nessie source where you want to run the SQL query.

Example: `nessieSource1`

Parameters of the `<Nessie source>` Object

type Body String Optional

The type of Nessie source object where you want to run the SQL query.

Enum: BRANCH, TAG, COMMIT

Example: BRANCH

value Body String Optional

The branch or tag name or commit hash in the Nessie source on which you want to run the SQL query.

Example: testing

Example Request Using Only the SQL Parameter

```
curl -X POST 'https://{hostname}/api/v3/sql' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "sql": "SELECT * FROM Samples.\"samples.dremio.com\".\"SF weather 2018-2019.csv\"
}'
```

Example Request Using Optional Parameters

```
curl -X POST 'https://{hostname}/api/v3/sql' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "sql": "SELECT * FROM \"SF weather 2018-2019.csv\"",
  "context": [
    "Samples",
    "samples.dremio.com"
  ],
  "references": {
    "nessieSource1": {
      "type": "BRANCH",
      "value": "testing"
    },
    "nessieSource2": {
      "type": "TAG",
      "value": "Test commit"
    },
    "nessieSource3": {
      "type": "COMMIT",
      "value": "7a5edb57e035f52beccfab632cea070514eb8b773f616aaeaf668e2f0be8f10d"
    }
  }
}'
```

Example Response

```
{
  "id": "2f067496-7cf0-a70e-0222-34d53a5dc800"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

Source](/25.x/reference/api/source)[Next

Token](/25.x/reference/api/token/)

Submitting an SQL Query

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/token/>

Version: 25.x

On this page

Token Enterprise

Use the Token API to manage personal access tokens associated with Dremio users.

Deleting All Tokens

Delete all personal access tokens for the user sending the API request.

Method and URL

```
DELETE /api/v3/token
```

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/token' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Deleting a Single Token

Delete the specified personal access token for the specified user.

Method and URL

```
DELETE /api/v3/user/{userName}/token/{id}
```

Parameters

userName Path String

Name of the Dremio user whose personal access token you want to delete.

Example: exampleuser1

id Path String (UUID)

Token ID for the personal access token you want to delete.

Example: 3eca2b0e-d122-48d6-9a97-f81b18db2380

Example Request

```
curl -X DELETE \
'https://{hostname}/api/v3/user/exampleuser1/token/3eca2b0e-d122-48d6-9a97-f81b18db2380' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

SQL](/25.x/reference/api/sql/)[Next

User](/25.x/reference/api/user/)

Deleting All Tokens

Response Status Codes

Deleting a Single Token

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/user/>

Version: 25.x

On this page

User Enterprise

Use the User API to manage Dremio users, their privileges, and their personal access tokens.

User Object

```
{
  "@type": "EnterpriseUser",
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "tag": "EuCNt1nnvdI=",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ],
  "source": "local",
  "active": true
}
```

User Attributes

@type String

Type of user.

Enum: EnterpriseUser, User

Example: EnterpriseUser

id String (UUID)

Unique identifier of the user.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

name String

Username of the Dremio user account.

Example: dremio

firstName String

User's first name.

Example: Dre

lastName String

User's last name.

Example: Mio

email String

User's email address. If the user is managed with the Dremio Okta application, email is the primary email address in the user's Okta profile. If the user is managed with Microsoft Entra ID, email is the work email address in the user's Microsoft Entra ID profile.

Example: user@dremio.com

tag String

Unique identifier of the user version. Dremio changes the tag whenever the user changes and uses the tag to ensure that PUT requests apply to the most recent version of the user.

Example: EuCNT1nnvdl=

roles Array of Object

Information about the local and referenced external roles to which the user belongs.

Example: [{"id": "8ac1bbca-479c-4c47-87e9-7f946f665c13", "name": "PUBLIC", "type": "SYSTEM"}, {"id": "43dce6d7-40ff-4afa-9901-71c30eb92744", "name": "ADMIN", "type": "SYSTEM"}]

source String

Information about how the user was created.

``external``: User was imported with an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

``local``: User was created manually in the Dremio user interface (UI) or with the User API.

Example: local

active Boolean

If the user account is active in Dremio, the value is ``true``. Otherwise, the value is ``false``. The active value is set to ``true`` when the user is created and only changes if the user's status changes in external System for Cross-domain Identity Management (SCIM) provisioning. When the user is activated in the SCIM application, Dremio sets the value to ``true``. When the user is deactivated in the SCIM application, Dremio sets the value to ``false``.

Example: true

Attributes of Objects in the ``roles`` Array

id String (UUID)

Unique identifier of the role.

Example: 43dce6d7-40ff-4afa-9901-71c30eb92744

name String

Name of the role.

Example: ADMIN

type String

Origin of the role.

``INTERNAL``: Role was created in the Dremio user interface (UI) or with the Role API.

``EXTERNAL``: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

``SYSTEM``: Role was predefined in Dremio.

Example: SYSTEM

Creating a User

Create a Dremio user.

Parameters

name Body String

Username for the Dremio user account. The name must be unique and cannot be updated after the user is created.

Example: dremio

firstName Body String Optional

User's first name.

Example: Dre

lastName Body String Optional

User's last name.

Example: Mio

email Body String Optional

User's email address.

Example: user@dremio.com

roles Body Array of Object Optional

Information about the roles to which the user should be assigned. All users are assigned to the PUBLIC role by default.

Example: [{ "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13", "name": "PUBLIC", "type": "SYSTEM"}, { "id": "43dce6d7-40ff-4afa-9901-71c30eb92744", "name": "ADMIN", "type": "SYSTEM"}]

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role.

Example: 43dce6d7-40ff-4afa-9901-71c30eb92744

name Body String

Name of the role. All users are assigned to the PUBLIC role by default.

Example: ADMIN

type Body String Optional

Origin of the role.

~INTERNAL~: Role was created in the Dremio user interface (UI) or with the Role API.

~EXTERNAL~: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

~SYSTEM~: Role was predefined in Dremio.

Example: SYSTEM

Example Request

```
curl -X POST 'https://{hostname}/api/v3/user' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ]
}'
```

Example Response

```
{
  "@type": "EnterpriseUser",
  "id": "b9dbecb7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "tag": "EuCNt1nnvdI=",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
      "name": "ADMIN",

```

```
    "type": "SYSTEM"
  }
],
"source": "external",
"active": true
}
```

Response Status Codes

200 OK

204 No Content

400 Bad Request

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Retrieving a User by ID

Retrieve a specific user by the user's ID.

Method and URL

```
GET /api/v3/user/{id}
```

Parameters

id Path String

Unique identifier of the user you want to retrieve.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example Request

```
curl -X GET 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "@type": "EnterpriseUser",
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
```

```
{
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "tag": "EuCNt1nnvdI=",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ],
  "source": "local",
  "active": true
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieving a User by Name

Retrieve a specific user by the user's name.

Method and URL

```
GET /api/v3/user/by-name/{name}
```

Parameters

name Path String

User name of the user you want to retrieve. User names are case-insensitive. If the user name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example Request

```
curl -X GET 'https://{hostname}/api/v3/user/by-name/dremio' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

When you retrieve a user by name, the response is an abbreviated user object that does not include the @type, roles, or source attributes:

Example Response

```
{
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "tag": "EuCNt1nnvdI=",
  "active": true
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Updating a User

Update the specified user.

Method and URL

```
PUT /api/v3/user/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user you want to update.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

id Body String (UUID)

Unique identifier of the user you want to update.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag Body String

Unique identifier of the user version to update. Dremio uses the tag to ensure that you are updating the most recent version of the user.

Example: BNGRmgfEnDg=

name Body String

Name of the user.

Example: dremio

firstName Body String Optional

User's first name.

Example: Dre

lastName Body String Optional

User's last name.

Example: Mio

email Body String Optional

User's email address.

Example: user@dremio.com

roles Body String Optional

Information about the roles to which the user should be assigned. All users are assigned to the PUBLIC role by default.

Example: [{"id": "8ac1bbca-479c-4c47-87e9-7f946f665c13","name": "PUBLIC","type": "SYSTEM"}, {"id": "43dce6d7-40ff-4afa-9901-71c30eb92744","name": "ADMIN","type": "SYSTEM"}]

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role.

Example: 43dce6d7-40ff-4afa-9901-71c30eb92744

name Body String

Name of the role. All users are assigned to the PUBLIC role by default.

Example: VIEWER

type Body String Optional

Origin of the role.

-

`INTERNAL`: Role was created in the Dremio user interface (UI) or with the Role API.

`EXTERNAL`: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

`SYSTEM`: Role was predefined in Dremio.

Enum: SYSTEM, INTERNAL, EXTERNAL

Example: INTERNAL

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "tag": "EuCNt1nnvdI=",
  "firstName": "Dremio",
  "lastName": "User",
  "email": "user@dremio.com",
  "roles": [
    {
      "id": "2f498015-9211-4b15-8fc0-493628ae7b6e",
      "name": "VIEWER"
    }
  ]
}'
```

Example Response

```
{
  "@type": "EnterpriseUser",
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "firstName": "Dremio",
  "lastName": "User",
  "email": "user@dremio.com",
  "tag": "BE1LYg3cmAk=",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "2f498015-9211-4b15-8fc0-493628ae7b6e",
      "name": "VIEWER",
      "type": "INTERNAL"
    }
  ],
  "source": "external",
}
```



```
"active": true
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Deleting a User

Delete the specified user.

Method and URL

```
DELETE /api/v3/user/{id}?version={tag}
```

Parameters

id Path String (UUID)

Unique identifier of the user that you want to delete. You can only delete users that are **not** currently logged in to Dremio.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

version Query String

Unique identifier of the user version to delete. The version value is the user's tag, which you can find in the response for a request to [Retrieve a User by ID](#) or [Retrieve a User by Name](#). Dremio uses the version value to ensure that you are deleting the most recent version of the user. If you provide an incorrect tag, the response includes an error message that lists the correct tag for the specified user ID.

Example: ?version=BE1LYg3cmAk=

Example Request

```
curl -X DELETE \
'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43?version=BE1LYg3cmAk=' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Token](/25.x/reference/api/token/)[Next

User Privileges](/25.x/reference/api/user/privilege)

User Attributes

Creating a User

Parameters

Response Status Codes

Retrieving a User by ID

Parameters

Response Status Codes

Retrieving a User by Name

Parameters

Response Status Codes

Updating a User

Parameters

Response Status Codes

Deleting a User

Parameters

Response Status Codes

Source:

<https://docs.dremio.com/25.x/reference/api/wlm>

/

Version: 25.x

Workload Management Enterprise

Use the Workload Management API to manage cluster resources and workloads by defining queues that have specific characteristics like memory limits, CPU priority, and queueing and runtime timeouts. You can also define rules that determine how to assign queries to specific queues.

See Workload Management for more information about Dremio's workload-management features.

Was this page helpful?

[Previous

User Tokens](/25.x/reference/api/user/token)[Next

Queue](/25.x/reference/api/wlm/queue)

Source:
<https://docs.dremio.com/25.x/reference/api/reflections/reflection-summary/>

Version: 25.x

On this page

Reflection Summary Enterprise

Use the Reflection API to retrieve a Reflection summary that includes the raw and aggregation Reflections for the Dremio instance.

Reflection summary objects are different from Reflection objects. Reflection summaries do not include certain attributes that define the Reflection, like the display, dimension, measure, sort, and partition attributes. Reflection summaries do include several attributes that do not appear in Reflection objects, like `datasetType`, `datasetPath`, and counts and links for considered, matched, and chosen jobs.

Reflection Summary Object

```
{
  "data": [
    {
      "createdAt": "2022-07-05T19:19:40.244Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
```

```

"totalSizeBytes": 9272,
"datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
"datasetType": "VIRTUAL_DATASET",
"datasetPath": [
  "Samples",
  "samples.dremio.com",
  "NYC-taxi-trips"
],
"status": {
  "configStatus": "OK",
  "refreshStatus": "MANUAL",
  "availabilityStatus": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "refreshMethod": "FULL",
  "failureCount": 0,
  "lastDataFetchAt": "2023-01-13T19:05:03.532Z",
  "expiresAt": "3022-05-16T19:46:02.342Z",
  "lastRefreshDurationMillis": 46387
},
"consideredCount": 202,
"matchedCount": 45,
"chosenCount": 5,
"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"isArrowCachingEnabled": false,
"isCanView": true,
"isCanAlter": true,
"isEnabled": true
},
{
  "createdAt": "2023-01-26T23:27:04.281Z",
  "updatedAt": "2023-01-26T23:27:04.281Z",
  "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
  "reflectionType": "RAW",
  "name": "Raw Reflection",
  "currentSizeBytes": 0,
  "outputRecords": -1,
  "totalSizeBytes": 0,
  "datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
  "datasetType": "PHYSICAL_DATASET",
  "datasetPath": [
    "@dremio",
    "test"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "GIVEN_UP",

```

```

    "availabilityStatus": "NONE",
    "combinedStatus": "FAILED",
    "refreshMethod": "NONE",
    "failureCount": 3,
    "lastFailureMessage": "The Default engine is not online.",
    "lastDataFetchAt": null,
    "expiresAt": null,
    "lastRefreshDurationMillis": -1
  },
  "consideredCount": 0,
  "matchedCount": 0,
  "chosenCount": 0,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:19.030Z",
  "updatedAt": "2023-01-13T19:50:19.030Z",
  "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
  "reflectionType": "AGGREGATION",
  "name": "dataset991_agg991",
  "currentSizeBytes": 9273,
  "outputRecords": 51,
  "totalSizeBytes": 9273,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
    "expiresAt": "3022-05-16T19:46:02.342Z",
    "lastRefreshDurationMillis": 11697
  },
  "consideredCount": 60,
  "matchedCount": 9,

```

```

    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:17.714Z",
    "updatedAt": "2023-01-13T19:50:17.714Z",
    "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
    "reflectionType": "RAW",
    "name": "dataset991_raw991",
    "currentSizeBytes": 818790,
    "outputRecords": 29467,
    "totalSizeBytes": 818790,
    "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
    "datasetType": "VIRTUAL_DATASET",
    "datasetPath": [
      "temp",
      "dataset991"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "MANUAL",
      "availabilityStatus": "AVAILABLE",
      "combinedStatus": "CAN_ACCELERATE",
      "refreshMethod": "FULL",
      "failureCount": 0,
      "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
      "expiresAt": "3022-05-16T19:46:02.747Z",
      "lastRefreshDurationMillis": 16666
    },
    "consideredCount": 54,
    "matchedCount": 37,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,

```

```

    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  }
],
"nextPageToken": "CiQxNjc0MjhIYi03OTM2LTRlYtItYTfmYi0yM2IxYWM2ZTk0NTQSAA==",
"isCanAlterReflections": true
}

```

Reflection Summary Attributes

data Array of Object

List of Reflection-summary objects for each Reflection in the Dremio instance.

nextPageToken String

Opaque string to pass for the `pageToken` query parameter in the next request to retrieve the next set of results. If nextPageToken is not included in the response, all available resources have been returned.

Example: CiQxNjc0MjhIYi03OTM2LTRlYtItYTfmYi0yM2IxYWM2ZTk0NTQSAA==

isCanAlterReflections Boolean

If the current user has project-level privileges to alter Reflections, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of Objects in the `data` Array

createdAt String

Date and time that the Reflection was created, in UTC format.

Example: 2022-07-05T19:19:40.244Z

updatedAt String

Date and time that the Reflection was last updated, in UTC format.

Example: 2023-01-13T19:46:01.313Z

id String (UUID)

Unique identifier of the Reflection.

Example: 27077c03-ae49-454c-a7bb-a9a8b5eca224

reflectionType String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio UI, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: NYC_taxi_agg

currentSizeBytes Integer

Data size of the latest Reflection job (if one exists), in bytes.

Example: 9272

outputRecords Integer

Number of records returned for the latest Reflection.

Example: 51

totalSizeBytes Integer

Data size of all Reflection jobs that have not been pruned (if any exist), in bytes.

Example: 9272

datasetId String

Unique identifier of the anchor dataset that is associated with the Reflection.

Example: fa7c487f-9550-474e-8a41-4826564c6b09

datasetType String

Type for the anchor dataset that is associated with the Reflection. If the anchor dataset is a table, the type is `PHYSICAL_DATASET`. If the anchor dataset is a view, the type is `VIRTUAL_DATASET`.

Enum: PHYSICAL_DATASET, VIRTUAL_DATASET

Example: VIRTUAL_DATASET

datasetPath String

Path to the anchor dataset that is associated with the Reflection within Dremio, expressed in an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the dataset itself as the last item in the array.

Example: ["Samples","samples.dremio.com","NYC Taxi Trips"]

status Object

Information about the status of the Reflection.

Example: {\n "configStatus": "OK",\n "refreshStatus": "MANUAL",\n "availabilityStatus":


```
"AVAILABLE",\n "combinedStatus": "CAN\\_ACCELERATE",\n "refreshMethod": "FULL",\n "failureCount": 0,\n "lastDataFetchAt": "2023-01-13T19:05:03.532Z",\n "expiresAt":  
"3022-05-16T19:46:02.342Z",\n "lastRefreshDurationMillis": 46387\n }
```

consideredCount Integer

Number of jobs that considered the Reflection during planning.

Example: 202

matchedCount Integer

Number of jobs that matched the Reflection during planning.

Example: 45

chosenCount Integer

Number of jobs accelerated by the Reflection.

Example: 5

consideredJobsLink String

Link to list of considered jobs for the Reflection.

Example:

```
/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%  
22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATI  
ON%22%5D%7D
```

matchedJobsLink String

Link to list of matched jobs for the Reflection.

Example:

```
/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%  
22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATI  
ON%22%5D%7D
```

chosenJobsLink String

Link to list of chosen jobs for the Reflection.

Example:

```
/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%  
22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATI  
ON%22%5D%7D
```

isArrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, the value is `true`. Otherwise, the value is `false`.

Example: false

isCanView Boolean

If you can view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

isCanAlter Boolean

If you can create, edit, and view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

isEnabled Boolean

If the Reflection is available for accelerating queries, the value is ``true``. Otherwise, the value is ``false``.

Example: true

Attributes of the ``status`` Object

configStatus String

Status of the Reflection configuration. If the value is ``OK``, the Reflection configuration is free of errors. If the value is ``INVALID``, the Reflection configuration contains one or more errors.

Enum: OK, INVALID

Example: OK

refreshStatus String

Status of the Reflection refresh.

``GIVEN_UP``: Dremio attempted to refresh the Reflection multiple times, but each attempt has failed and Dremio will not make further attempts.

``MANUAL``: Refresh period is set to 0, so you must use the Dremio UI to manually refresh the Reflection.

``RUNNING``: Dremio is currently refreshing the Reflection.

``SCHEDULED``: The Reflection refreshes according to a schedule.

``ON_DATA_CHANGES``: All of the Reflection's underlying tables are in Iceberg format, and the Reflection refreshes automatically if new snapshots are created after an update to the underlying tables.

Enum: GIVEN_UP, MANUAL, RUNNING, SCHEDULED, ON_DATA_CHANGES

Example: MANUAL

availabilityStatus String

Status of the Reflection's availability for accelerating queries.

Enum: NONE, INCOMPLETE, EXPIRED, AVAILABLE

Example: AVAILABLE

combinedStatus String

Status of the Reflection based on a combination of config, refresh, and availability.

˘CAN_ACCELERATE˘: The Reflection is fully functional.

˘CAN_ACCELERATE_WITH_FAILURES˘: The most recent refresh failed to obtain a status, but Dremio still has a valid materialization.

˘CANNOT_ACCELERATE_INITIALIZING˘: The Reflection is currently being loaded into the materialization cache. During this time, the Reflection is unable to accelerate queries.

˘CANNOT_ACCELERATE_MANUAL˘: The Reflection is unable to accelerate any queries, and the ˘Never Refresh˘ option is selected for the refresh policy.

˘CANNOT_ACCELERATE_SCHEDULED˘: The Reflection is currently unable to accelerate any queries, but it has been scheduled for a refresh at a future time.

˘DISABLED˘: The Reflection has been manually disabled.

˘EXPIRED˘: The Reflection has expired and cannot be used.

˘FAILED˘: The attempt to refresh the Reflection has failed, typically three times in a row. The Reflection is still usable.

˘INVALID˘: The Reflection is invalid because the underlying dataset has changed.

˘INCOMPLETE˘: One or more pseudo-distributed file system (PDFS) nodes that contain materialized files are down (PDFS is supported for v21 and earlier). Only partial data is available. Configurations that use the Hadoop Distributed File System (HDFS) to store Reflections should not experience incomplete status.

˘REFRESHING˘: The Reflection is currently being refreshed.

Example: CAN_ACCELERATE

refreshMethod String

The method used for the most recent refresh of the Reflection. For new Reflections, the value is ˘NONE˘ until planned. For more information, read [Refreshing Reflections](#).

Enum: NONE, FULL, INCREMENTAL

Example: FULL

failureCount Integer

Number of times that an attempt to refresh the Reflection failed.

Example: 0

lastFailureMessage String

The error message from the last failed Reflection refresh. If the refresh of a Reflection never fails or succeeds after a failure, this attribute does not appear.

Example: "The Default engine is not online."

lastDataFetchAt String

Date and time that the Reflection data was last refreshed, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetchAt value is `1969-12-31T23:59:59.999Z`.

Example: 2023-01-13T19:05:03.532Z

expiresAt String

Date and time that the Reflection expires, in UTC format. If the Reflection is running, failing, or disabled, the expiresAt value is `1969-12-31T23:59:59.999Z`.

Example: 3022-05-16T19:46:02.342Z

lastRefreshDurationMillis Integer

Duration of the most recent refresh for the Reflection. In milliseconds.

Example: 46387

Retrieving a Reflection Summary

Retrieve a summary of the raw and aggregation Reflections in the Dremio instance.

Method and URL

```
GET /api/v3/reflection-summary
```

Parameters

pageToken Query String Optional

Token for retrieving the next page of Reflection summary results. If the Dremio instance has more Reflection summary results than the maximum per page (default 50), the response includes a nextPageToken after the data array. Use the nextPageToken value in your request URL as the pageToken value. Do not change any other query parameters included in the request URL when you use pageToken. For more information, read [pageToken Query Parameter](#).

maxResults Query Integer Optional

Maximum number of Reflection summaries to return in the response. Maximum valid value is `100`. Default is `50`. For more information, read [maxResults Query Parameter](#).

filter Query Object Optional

Filters for Reflection name, dataset name, availability status, and refresh status. Value is a URL-encoded string that represents a JSON object. The JSON object specifies the attributes to filter on and the values to match for each attribute. Available filter attributes:

reflectionType: `RAW`, `AGGREGATION` (array of string)

`refreshStatus`: ``GIVEN_UP`, `MANUAL`, `RUNNING`, `SCHEDULED`, `ON_DATA_CHANGES`` (array of string)

`availabilityStatus`: ``NONE`, `INCOMPLETE`, `EXPIRED`, `AVAILABLE`` (array of string)

`configStatus`: ``OK`, `INVALID`` (array of string)

`enabledFlag`: ``true`, `false`` (Boolean)

`reflectionNameOrDatasetPath`: full or partial Reflection name or dataset path; case insensitive (string)

`reflectionIds`: IDs of Reflections to retrieve (array of string); must be used alone, with no other filters or query parameters

For more information, read [filter Query Parameter](#).

`orderBy` Query String Optional

Organize the response in ascending (default) or descending order by `reflectionName`, `datasetName`, or `reflectionType`. To specify descending order, precede the `orderBy` value with a ``-`` character. For more information, read [orderBy Query Parameter](#).

Example Request Without Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary'
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header "Content-Type: application/json"
```

Example Response

```
{
  "data": [
    {
      "createdAt": "2023-01-13T19:46:01.313Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
      "totalSizeBytes": 9272,
      "datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
      "datasetType": "VIRTUAL_DATASET",
      "datasetPath": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "status": {
        "configStatus": "OK",
        "refreshStatus": "MANUAL",
        "availabilityStatus": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "refreshMethod": "FULL",
```

```

    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
    "expiresAt": "3022-05-16T19:46:02.342Z",
    "lastRefreshDurationMillis": 46387
  },
  "consideredCount": 202,
  "matchedCount": 45,
  "chosenCount": 5,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
},
{
  "createdAt": "2023-01-26T23:27:04.281Z",
  "updatedAt": "2023-01-26T23:27:04.281Z",
  "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
  "reflectionType": "RAW",
  "name": "Raw Reflection",
  "currentSizeBytes": 0,
  "outputRecords": -1,
  "totalSizeBytes": 0,
  "datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
  "datasetType": "PHYSICAL_DATASET",
  "datasetPath": [
    "@dremio",
    "test"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "GIVEN_UP",
    "availabilityStatus": "NONE",
    "combinedStatus": "FAILED",
    "refreshMethod": "NONE",
    "failureCount": 3,
    "lastDataFetchAt": null,
    "expiresAt": null,
    "lastRefreshDurationMillis": -1
  },
  "consideredCount": 0,
  "matchedCount": 0,
  "chosenCount": 0,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",

```

```

"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",

"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:19.030Z",
  "updatedAt": "2023-01-13T19:50:19.030Z",
  "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
  "reflectionType": "AGGREGATION",
  "name": "dataset991_agg991",
  "currentSizeBytes": 9273,
  "outputRecords": 51,
  "totalSizeBytes": 9273,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
    "expiresAt": "3022-05-16T19:46:02.342Z",
    "lastRefreshDurationMillis": 11697
  },
  "consideredCount": 60,
  "matchedCount": 9,
  "chosenCount": 0,

"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",

"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",

"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
},

```

```

{
  "createdAt": "2023-01-13T19:50:17.714Z",
  "updatedAt": "2023-01-13T19:50:17.714Z",
  "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
  "reflectionType": "RAW",
  "name": "dataset991_raw991",
  "currentSizeBytes": 818790,
  "outputRecords": 29467,
  "totalSizeBytes": 818790,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
    "expiresAt": "3022-05-16T19:46:02.747Z",
    "lastRefreshDurationMillis": 16666
  },
  "consideredCount": 54,
  "matchedCount": 37,
  "chosenCount": 0,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
}
},
"nextPageToken": "CiQxNjc0Mjhlyi030TM2LTRlYtYTFmYi0yM2IxYWM2ZTk0NTQSA==",
"isCanAlterReflections": true
}

```

This endpoint supports query parameters that you can add to the request URL to include only specific types of Reflections in the Reflection summary, specify the maximum number of results to return, and sort the response to list Reflections in ascending or descending order.

For example, to order the Reflections within the summary in ascending order by reflectionName, add `?orderBy=reflectionName` to the request URL. For descending order, add a `-` character before the attribute name: `?orderBy=-reflectionName`.

In the same request, you can add the `filter` query parameter to retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths. The JSON object for such a filter would look like this:

Example JSON Object for Filter

```
{
  "reflectionType": ["RAW"],
  "refreshStatus": ["MANUAL", "SCHEDULED"],
  "enabledFlag": true,
  "reflectionNameOrDatasetPath": "samples.dremio.com"
}
```

However, to use the JSON object in the request URL, you must convert it to URL-encoded JSON, which looks like this:

Example JSON Object in URL-Encoded JSON

```
%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D
```

Here is an example request URL that includes both the `orderBy` and `filter` query parameters:

Example Request with orderBy and filter Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName&filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

For this request, the Reflection summary in the response will include only raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths, and the Reflections will be listed in ascending order by reflectionName.

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Recommendations](/25.x/reference/api/reflections/reflection-recommendations)[Next

Role](/25.x/reference/api/roles/)

Reflection Summary Attributes

Retrieving a Reflection Summary

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/source/>

Version: 25.x

On this page

Source

Use the Catalog API to retrieve information about [sources](#) and the child objects they contain, as well as to create, update, and delete sources.

note

Dremio supports a number of different source types. Each source type has the same parameters **except** for the parameters within the ``config`` object. The available parameters in the ``config`` object are different for each source type. The examples on this page use an Amazon S3 source to demonstrate the available requests and responses for sources. Read [Source Configuration](#) for information about the available parameters in the ``config`` object for each supported source type.

Source Object

```
{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
```

```

    "logs_east-1",
    "logs_west-1"
  ],
  "requesterPays": false,
  "enableFileStatusCheck": true,
  "defaultCtasFormat": "ICEBERG",
  "isPartitionInferenceEnabled": false,
  "credentialType": "ACCESS_KEY"
},
"id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
"tag": "T0/Zr1FOY3A=",
"type": "S3",
"name": "AWS-S3_testgroup",
"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLs": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"accelerationActivePolicyType": "NEVER",
"accelerationRefreshSchedule": "",
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],

```

```

    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",
  "TRUNCATE",
  "DELETE",
  "UPDATE",
  "EXECUTE",
  "CREATE_SOURCE",

```

```

    "ALL"
  ],
  "checkTableAuthorizer": true,
  "owner": {
    "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
    "ownerType": "USER"
  },
  "accelerationRefreshOnDataChanges": false
}

```

Source Attributes

entityType String

Type of the catalog object. For sources, the entityType is `source`.

Example: source

config Object

Configuration settings for the source. The available parameters in the config object are different for different source types. For more information, read [Source Configuration](#).

Example: `{ "accessKey": "EXAMPLE78HT89VS4YJEL", "accessSecret": "$DREMIO_EXISTING_VALUE$", "secure": true, "rootPath": "/", "enableAsync": true, "compatibilityMode": false, "isCachingEnabled": true, "maxCacheSpacePct": 100, "requesterPays": false, "enableFileStatusCheck": true, "defaultCtasFormat": "ICEBERG", "isPartitionInferenceEnabled": false, "credentialType": "ACCESS_KEY" }`

id String (UUID)

Unique identifier of the source.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

tag String

Unique identifier of the version of the source. Dremio changes the tag whenever the source changes and uses the tag to ensure that PUT requests apply to the most recent version of the source.

Example: T0/Zr1FOY3A=

type String

Type of source.

Enum: ADL, ADX, AMAZONELASTIC, AWSGLUE, AZURE_STORAGE, DB2, DREMIOTODREMIO, ELASTIC, GCS, HDFS, HIVE, HIVE3, MONGO, MSSQL, MYSQL, NAS, NESSIE, ORACLE, POSTGRES, REDSHIFT, S3, SNOWFLAKE, SYNAPSE, TERADATA

Example: S3

name String

Name of the source.

Example: AWS-S3_testgroup

createdAt String

Date and time that the source was created, in UTC format.

Example: 2023-02-17T14:32:20.640Z

metadataPolicy Object

Information about the metadata policy for the source.

Example: { "authTTLMS": 86400000, "namesRefreshMS": 3600000, "datasetRefreshAfterMS": 3600000, "datasetExpireAfterMS": 10800000, "datasetUpdateMode": "PREFETCH_QUERIED", "deleteUnavailableDatasets": true, "autoPromoteDatasets": false }

accelerationGracePeriodMS Integer

Maximum age allowed for Reflection data used to accelerate queries on datasets in the source, in milliseconds. Default is `0`. For more information, read [Setting the Expiration Policy for Reflections](#).

Example: 10800000

accelerationRefreshPeriodMS Integer

Refresh period for the data in all Reflections on datasets in the source, in milliseconds. Default is `0`.

Example: 3600000

accelerationNeverExpire Boolean

Option to set an expiration for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from expiring and to override the `accelerationGracePeriodMS` setting.

Example: false

accelerationNeverRefresh Boolean

Option to set a refresh for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from refreshing and to override the `accelerationRefreshPeriodMS` setting.

accelerationActivePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `accelerationNeverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: The Reflections are refreshed at the end of every period that is defined by accelerationRefreshPeriodMS.

`^SCHEDULE^`: The Reflections are refreshed according to the schedule that is set by `accelerationRefreshSchedule`.

`accelerationRefreshSchedule` String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed.

Field	Allowed Values	Allowed Special Characters
-------	----------------	----------------------------

---	---	---
-----	-----	-----

Second	0	N/A
--------	---	-----

Minute	0-59	N/A
--------	------	-----

Hour	0-23	N/A
------	------	-----

Day of month	N/A	* ?
--------------	-----	------

Month	N/A	* ?
-------	-----	------

Days of week	1-7 or SUN-SAT	, - * ?
--------------	----------------	----------

Special Character	Description
-------------------	-------------

---	---
-----	-----

*	Used to specify all values for a field. For <code>Day of month</code> , specifies every day of the month. For <code>Month</code> , specifies every month. For <code>Days of week</code> , specifies every day of the week.
----	--

?	Equivalent to *.
---	-------------------

,	Used to specify two or more days in the <code>Days of week</code> field. For example, <code>MON,WED,FRI</code> .
---	--

-	Used to specify ranges in the <code>Days of week</code> field. For example, <code>1-3</code> is equivalent to <code>Sunday, Monday, and Tuesday</code> .
---	--

Examples:

`0 0 0 * * ?`` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

`children` Array of Object

Information about each catalog object in the source.

Example:

```
[{"id": "dremio:/AWS-S3_testgroup/archive.dremio.com", "path": ["AWS-S3_testgroup", "archive.dremio.com"], "type": "CONTAINER", "containerType": "FOLDER"}, {"id": "dremio:/AWS-S3_testgroup/logs_east-1", "path": ["AWS-S3_testgroup", "logs_east-1"], "type": "CONTAINER", "containerType": "FOLDER"}, {"id": "dremio:/AWS-S3_testgroup/logs_west-1", "path": ["AWS-S3_testgroup", "logs_west-1"], "type": "CONTAINER", "containerType": "FOLDER"}]
```

`allowCrossSourceSelection` Boolean

If the source is available for queries that can select from multiple sources, set to `true``. Otherwise, set to `false`` (default).

Example: `false`

`disableMetadataValidityCheck` Boolean

To disable the check for expired metadata and require users to refresh manually, set to `true``. Otherwise, set to `false`` (default).

note

The `disableMetadataValidityCheck` attribute is not supported by default. Contact Dremio Support to enable it.

Example: `false`

accessControlList Object

Enterprise only. Information about users and roles with access to the source and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The `accessControlList` array is empty if source-specific access control privileges are not set.

Example: `{"users": [{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65", "permissions": ["VIEW_REFLECTION", "SELECT"]}], "roles": [{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"]}]}`

permissions Array of String

Enterprise-only. List of the privileges that you have on the source. Empty unless the request URL includes the ``permissions`` query parameter. For more information, read [Privileges](#).

Example:

`["READ", "WRITE", "ALTER_REFLECTION", "SELECT", "ALTER", "VIEW_REFLECTION", "MODIFY", "MANAGE_GRANTS", "CREATE_TABLE", "DROP", "EXTERNAL_QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE_SOURCE", "ALL"]`

checkTableAuthorizer Boolean

Not used. Has the value ``true``.

Example: `true`

owner Object

Information about the source's owner.

Example: `{"ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365", "ownerType": "USER"}`

accelerationRefreshOnDataChanges Boolean

If Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update, ``true``. Otherwise, ``false``.

Attributes of the ``config`` Object

The ``config`` object attributes vary for different source types. Read [Source Configuration](#) for information about the available parameters in the ``config`` object for each supported source type.

Attributes of the ``metadataPolicy`` Object

authTTLMs Integer

Length of time to cache the privileges that the user has on the source, in milliseconds. For example, if authTTLMs is set to `28800000` (8 hours), Dremio checks the user's permission status once every 8 hours. Default is `86400000` (24 hours). Minimum is `60000` (1 minute).

Example: 86400000

namesRefreshMs Integer

How often the source is refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetRefreshAfterMs Integer

How often the metadata in the source's datasets is refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetExpireAfterMs Integer

Maximum age allowed for the metadata in the source's datasets, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 10800000

datasetUpdateMode String

Approach Dremio uses for updating the metadata when updating datasets in the source.

`PREFETCH`: (deprecated) Dremio updates details for all datasets in a source.

`PREFETCH_QUERIED`: Dremio updates details for previously queried objects in a source.

Example: PREFETCH_QUERIED

deleteUnavailableDatasets Boolean

If Dremio removes dataset definitions from the source when the underlying data is unavailable, set to `true` (default). Otherwise, set to `false`.

Example: true

autoPromoteDatasets Boolean

If Dremio automatically formats files into tables when a user issues a query, set to `true`. Otherwise, set to `false` (default). Available only for datalake sources, such as Amazon S3 and Hive.

Example: false

Attributes of Objects in the `children` Array

id String

Unique identifier of the child catalog object. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: `dremio:/AWS-S3_testgroup/archive.dremio.com`

path Array of String

Path to the child catalog object within the source, expressed as an array. The path consists of the source, followed by the name of the folder, file, or dataset itself as the last item in the array.

Example: `["AWS-S3_testgroup","archive.dremio.com"]`

type String

Type of the catalog object.

Enum: CONTAINER, FILE, DATASET

Example: CONTAINER

containerType String

For catalog objects with the type ``CONTAINER``, the containerType is ``FOLDER``.

Example: FOLDER

datasetType String

For catalog objects with the type ``DATASET``, the type of dataset. If the dataset is from an external source such as PostgreSQL, the datasetType is ``DIRECT``. For tables, the datasetType is ``PROMOTED``. For views, the datasetType is ``VIRTUAL``.

Enum: DIRECT, PROMOTED, VIRTUAL

Example: VIRTUAL

Attributes of the ``accessControlList`` Object

users Array of Object

Enterprise only. List of users with access to the source and the specific privileges each user has.

Example: `[{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65","permissions": ["VIEW_REFLECTION","SELECT"]}]`

roles Array of Object

Enterprise only. List of roles whose members have access to the source and the specific privileges each role has.

Example: `[{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56","permissions": ["ALTER","CREATE_TABLE","DROP","INSERT","DELETE","UPDATE","TRUNCATE",`

"VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the source's owner.

Example: 4fb93af3-acc2-4b10-ad4b-64dd7070d365

ownerType String

Type of owner of the source.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise only. Unique identifier of the user or role with access to the source.

Example: ebe519ab-20e3-43ff-9b4c-b3ec590c7e65

permissions Array of String

Enterprise only. List of privileges the user or role has on the source. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["VIEW_REFLECTION","SELECT"]

Creating a Source

Create a new source.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object to create. For sources, the entityType is `source`.

Example: source

`config` Body Object

Configuration settings for the source. The available parameters in the config object are different for different source types. For more information, read [Source Configuration](#).

Example:

```
{ "accessKey": "EXAMPLE78HT89VS4YJEL", "accessSecret": "EXAMPLEe3bcpKnAwgJ2WBpX8d9kEdhMz24guiR7L", "secure": true, "rootPath": "/", "enableAsync": true, "compatibilityMode": false, "isCachingEnabled": true, "maxCacheSpacePct": 100, "requesterPays": false, "enableFileStatusCheck": true, "defaultCtasFormat": "ICEBERG", "isPartitionInferenceEnabled": false, "credentialType": "ACCESS_KEY" }
```

type Body String

Type of source to create.

Enum: ADL, ADX, AMAZONELASTIC, AWSGLUE, AZURE_STORAGE, DB2, DREMIOTODREMIO, ELASTIC, GCS, HDFS, HIVE, HIVE3, MONGO, MSSQL, MYSQL, NAS, NESSIE, ORACLE, POSTGRES, REDSHIFT, S3, SNOWFLAKE, SYNAPSE, TERADATA

Example: S3

name Body String

Name for the source. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Example: AWS-S3_testgroup

`metadataPolicy` Body Object Optional

Information about the metadata policy for the source.

Example:

```
{ "authTTLs": 86400000, "namesRefreshMs": 3600000, "datasetRefreshAfterMs": 3600000, "datasetExpireAfterMs": 10800000, "datasetUpdateMode": "PREFETCH_QUERIED", "deleteUnavailableDatasets": true, "autoPromoteDatasets": false }
```

accelerationGracePeriodMs Body Integer Optional

Maximum age to allow for Reflection data used to accelerate queries on datasets in the source, in milliseconds. Default is ``0``. For more information, read [Setting the Expiration Policy for Reflections](#).

Example: 10800000

accelerationRefreshPeriodMs Body Integer Optional

Refresh period to use for the data in all Reflections on datasets in the source, in milliseconds. Optional if you set `accelerationActivePolicyType` to ``PERIOD``. The default setting is ``3600000`` milliseconds or one hour.

Example: 3600000

accelerationNeverExpire Body Boolean Optional

Option to set an expiration for Reflections. Default setting is ``false``. Set to ``true`` to prevent Reflections from expiring and to override the ``accelerationGracePeriodMs``

setting.

Example: false

accelerationNeverRefresh Body Boolean Optional

Option to set a refresh for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from refreshing and to override the `accelerationRefreshPeriodMs` setting.

accelerationActivePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `accelerationNeverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by accelerationRefreshPeriodMs.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by accelerationRefreshSchedule.

accelerationRefreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed. Optional if you set accelerationActivePolicyType to `SCHEDULE`. The default accelerationRefreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
?	Equivalent to *.
,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

allowCrossSourceSelection Body Boolean Optional

If the source should be available for queries that can select from multiple sources, set to ``true``. Otherwise, set to ``false`` (default).

Example: false

`disableMetadataValidityCheck` Body Boolean Optional

To disable the check for expired metadata and require users to refresh manually, set to ``true``. Otherwise, set to ``false`` (default).

note

The `disableMetadataValidityCheck` parameter is not supported by default. Contact Dremio Support to enable it.

Example: false

`accelerationRefreshOnDataChanges` Body Boolean

To refresh Reflections on underlying tables that are in Iceberg format in the source when new snapshots are created after an update, ``true``. Otherwise, ``false``. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. For this option to take effect, the source must support Iceberg table format, the `accelerationNeverRefresh` parameter must be set to ``false``, and the `accelerationActivePolicyType` parameter must be set to either ``PERIOD`` or ``SCHEDULE``.

`accessControllist` Body Object Optional

Enterprise only. Information about users and roles that should have access to the source and the specific privileges each user or role should have. May include an array of users, an array of roles, or both, depending on the configured access and privileges.

Example:

```
{
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
}
```

Parameters of the ``config`` Object

The ``config`` object's parameters vary for different source types. Read [Source Configuration](#) for information about the available parameters in the ``config`` object for each supported source type.

Parameters of the ``metadataPolicy`` Object

`authTTLms` Body Integer

Length of time to cache the privileges that the user has on the source, in milliseconds. For example, if `authTTLms` is set to ``28800000`` (8 hours), Dremio checks the user's permission status once every 8 hours. Default is ``86400000`` (24 hours). Minimum is ``60000`` (1 minute).

Example: 86400000

namesRefreshMs Body Integer Optional

How often the source should be refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetRefreshAfterMs Body Integer Optional

How often the metadata in the source's datasets should be refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetExpireAfterMs Body Integer Optional

Maximum age to allow for the metadata in the source's datasets, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 10800000

datasetUpdateMode Body String Optional

Approach for Dremio to take for updating the metadata when updating datasets in the source.

`PREFETCH`: (deprecated) Dremio updates details for all datasets in a source.

`PREFETCH_QUERIED`: Dremio updates details for previously queried objects in a source.

Example: PREFETCH_QUERIED

deleteUnavailableDatasets Body Boolean Optional

If Dremio should remove dataset definitions from the source when the underlying data is unavailable, set to `true` (default). Otherwise, set to `false`.

Example: true

autoPromoteDatasets Body Boolean Optional

If Dremio should automatically format files into tables using default options when users issue queries, set to `true`. Otherwise, set to `false` (default). Available only for datalake sources, such as Amazon S3 and Hive.

Example: false

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the source and the specific privileges each user should have.

Example: `[{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65", "permissions":`

```
["VIEW\_REFLECTION","SELECT"]}]}
```

roles Body Array of Object Optional

List of roles whose members should have access to the source and the specific privileges each role should have.

Example: [{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions": ["ALTER","CREATE_TABLE","DROP","INSERT","DELETE","UPDATE","TRUNCATE","VIEW_REFLECTION","ALTER_REFLECTION","MODIFY","MANAGE_GRANTS","SELECT"]}]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Unique identifier of the user or role who should have access to the source.

Example: ebe519ab-20e3-43ff-9b4c-b3ec590c7e65

permissions Body Array of String Optional

List of privileges the user or role should have on the source. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["VIEW_REFLECTION","SELECT"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "EXAMPLEe3bcpKnAwgJ2WBpX8d9kEdhMz24guiR7L",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  }
}
```



```

},
"type": "S3",
"name": "AWS-S3_testgroup",
"metadataPolicy": {
  "authTTLMs": 864000000,
  "namesRefreshMs": 36000000,
  "datasetRefreshAfterMs": 36000000,
  "datasetExpireAfterMs": 108000000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 108000000,
"accelerationRefreshPeriodMs": 36000000,
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ]
},
"roles": [
  {
    "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
    "permissions": [
      "ALTER",
      "CREATE_TABLE",
      "DROP",
      "INSERT",
      "DELETE",
      "UPDATE",
      "TRUNCATE",
      "VIEW_REFLECTION",
      "ALTER_REFLECTION",
      "MODIFY",
      "MANAGE_GRANTS",
      "SELECT"
    ]
  }
]
}
}'

```

Example Response

```
{
```

```

"entityType": "source",
"config": {
  "accessKey": "EXAMPLE78HT89VS4YJEL",
  "accessSecret": "$DREMIO_EXISTING_VALUE$",
  "secure": true,
  "rootPath": "/",
  "enableAsync": true,
  "compatibilityMode": false,
  "isCachingEnabled": true,
  "maxCacheSpacePct": 100,
  "whitelistedBuckets": [
    "archive.dremio.com",
    "logs_east-1",
    "logs_west-1"
  ],
  "requesterPays": false,
  "enableFileStatusCheck": true,
  "defaultCtasFormat": "ICEBERG",
  "isPartitionInferenceEnabled": false,
  "credentialType": "ACCESS_KEY"
},
"state": {
  "status": "good",
  "suggestedUserAction": "",
  "messages": []
},
"id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
"tag": "T0/Zr1FOY3A=",
"type": "S3",
"name": "AWS-S3_testgroup",
"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLms": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
]

```

```

    },
    {
      "id": "dremio:/AWS-S3_testgroup/logs_east-1",
      "path": [
        "AWS-S3_testgroup",
        "logs_east-1"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "dremio:/AWS-S3_testgroup/logs_west-1",
      "path": [
        "AWS-S3_testgroup",
        "logs_west-1"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    }
  ],
  "allowCrossSourceSelection": false,
  "disableMetadataValidityCheck": false,
  "accessControlList": {
    "users": [
      {
        "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
        "permissions": [
          "VIEW_REFLECTION",
          "SELECT"
        ]
      }
    ]
  },
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"checkTableAuthorizer": true,

```

```
"owner": {
  "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
  "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": false
}
```

When you use the Catalog API to create a new source, the response includes a `state` object that describes the status of the source as shown in the example response above. The `state` object contains the following attributes:

status String

Status of the created source.

Enum: good, bad, warn

Example: good

suggestedUserAction String

Recommended action to take, if any, based on the status of the created source.

messages Array of String

Status message, if any, for the created source.

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieving a Source by ID

Retrieve a source and information about its contents by specifying the source's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the source that you want to retrieve.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the source has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "source",
  "config": {
```

```

"accessKey": "EXAMPLE78HT89VS4YJEL",
"accessSecret": "$DREMIO_EXISTING_VALUE$",
"secure": true,
"rootPath": "/",
"enableAsync": true,
"compatibilityMode": false,
"isCachingEnabled": true,
"maxCacheSpacePct": 100,
"whitelistedBuckets": [
  "archive.dremio.com",
  "logs_east-1",
  "logs_west-1"
],
"requesterPays": false,
"enableFileStatusCheck": true,
"defaultCtasFormat": "ICEBERG",
"isPartitionInferenceEnabled": false,
"credentialType": "ACCESS_KEY"
},
"id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
"tag": "T0/Zr1F0Y3A=",
"type": "S3",
"name": "AWS-S3_testgroup",
"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLms": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
  },

```

```

    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"checkTableAuthorizer": true,
"owner": {
  "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
  "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a Source by Path

Retrieve a source and information about its contents by specifying the source's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Name of the source that you want to retrieve. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: AWS-S3_testgroup

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the source has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/AWS-S3_testgroup' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
  "tag": "T0/Zr1FOY3A=",
  "type": "S3",
  "name": "AWS-S3_testgroup",
  "createdAt": "2023-02-17T14:32:20.640Z",
  "metadataPolicy": {
    "authTTLs": 86400000,
  }
}
```

```

"namesRefreshMs": 3600000,
"datasetRefreshAfterMs": 3600000,
"datasetExpireAfterMs": 10800000,
"datasetUpdateMode": "PREFETCH_QUERIED",
"deleteUnavailableDatasets": true,
"autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControllist": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ]
},
"roles": [

```

```

{
  "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
  "permissions": [
    "ALTER",
    "CREATE_TABLE",
    "DROP",
    "INSERT",
    "DELETE",
    "UPDATE",
    "TRUNCATE",
    "VIEW_REFLECTION",
    "ALTER_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "SELECT"
  ]
},
{
  "permissions": [],
  "checkTableAuthorizer": true,
  "owner": {
    "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
    "ownerType": "USER"
  },
  "accelerationRefreshOnDataChanges": false
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Updating a Source

Update the specified source.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the source to update.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

entityType Body String

Type of the catalog object to update. For sources, the entityType is `source`.

Example: source

config Body Object

Configuration settings for the source. The available parameters in the config object are different for different source types. For more information, read [Source Configuration](#).

Example:

```
{ "accessKey": "EXAMPLE78HT89VS4YJEL", "accessSecret": "EXAMPLEe3bcpKnAwgJ2WBpX8d9kEdhMz24guiR7L", "secure": true, "rootPath": "/", "enableAsync": true, "compatibilityMode": false, "isCachingEnabled": true, "maxCacheSpacePct": 100, "requesterPays": false, "enableFileStatusCheck": true, "defaultCtasFormat": "ICEBERG", "isPartitionInferenceEnabled": false, "credentialType": "ACCESS_KEY" }
```

id Body String

Unique identifier of the source to update.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

tag Body String

Unique identifier of the version of the source that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the source.

Example: T0/Zr1FOY3A=

type Body String

Type of the source that you want to update.

Enum: ADL, ADX, AMAZONELASTIC, AWSGLUE, AZURE_STORAGE, DB2, DREMIOTODREMIO, ELASTIC, GCS, HDFS, HIVE, HIVE3, MONGO, MSSQL, MYSQL, NAS, NESSIE, ORACLE, POSTGRES, REDSHIFT, S3, SNOWFLAKE, SYNAPSE, TERADATA

Example: S3

name Body String

Name of the source that you want to update.

Example: AWS-S3_testgroup

metadataPolicy Body Object Optional

Information about the metadata policy for the source.

Example:

```
{ "authTTLs": 86400000, "namesRefreshMs": 3600000, "datasetRefreshAfterMs": 3600000, "datasetExpireAfterMs": 3600000 }
```

```
10800000,"datasetUpdateMode": "PREFETCH_QUERIED","deleteUnavailableDatasets": true,"autoPromoteDatasets": false}
```

accelerationGracePeriodMs Body Integer Optional

Maximum age to allow for Reflection data used to accelerate queries on datasets in the source, in milliseconds. For more information, read [Setting the Expiration Policy for Reflections](#).

Example: 10800000

accelerationRefreshPeriodMs Body Integer Optional

Refresh period to use for the data in all Reflections on datasets in the source, in milliseconds. Default is `0`.

Example: 3600000

accelerationNeverExpire Body Boolean Optional

Option to set an expiration for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from expiring and to override the `accelerationGracePeriodMs` setting.

Example: false

accelerationNeverRefresh Body Boolean Optional

Option to set a refresh for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from refreshing and to override the `accelerationRefreshPeriodMs` setting.

accelerationActivePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `accelerationNeverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: The Reflections are refreshed at the end of every period that is defined by accelerationRefreshPeriodMs.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by accelerationRefreshSchedule.

accelerationRefreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed. Optional if you set accelerationActivePolicyType to `SCHEDULE`. The default accelerationRefreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0 N/A	
Minute	0-59 N/A	
Hour	0-23 N/A	

Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

| Special Character | Description |

| --- | --- |

| * | Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week. |

| ? | Equivalent to *.

| , | Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.

| - | Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

`allowCrossSourceSelection` Body Boolean Optional

If the source should be available for queries that can select from multiple sources, set to `true`. Otherwise, set to `false` (default).

Example: false

`disableMetadataValidityCheck` Body Boolean Optional

To disable the check for expired metadata and require users to refresh manually, set to `true`. Otherwise, set to `false` (default).

note

The `disableMetadataValidityCheck` parameter is not supported by default. Contact Dremio Support to enable it.

Example: false

`accelerationRefreshOnDataChanges` Body Boolean

To refresh Reflections on underlying tables that are in Iceberg format in the source when new snapshots are created after an update, `true`. Otherwise, `false`. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. For this option to take effect, the source must support Iceberg table format, the `accelerationNeverRefresh` parameter must be set to `false`, and the `accelerationActivePolicyType` parameter must be set to either `PERIOD` or `SCHEDULE`.

`accessControlList` Body String Optional

Enterprise only. Information about users and roles that should have access to the source and the specific privileges each user or role should have. May include an array

of users, an array of roles, or both, depending on the configured access and privileges. To keep existing accessControlList settings while making other updates, duplicate the existing accessControlList object in the PUT request.

Example:

```
{
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": ["VIEW_REFLECTION", "SELECT"]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"]
    }
  ]
}
```

Parameters of the `config` Object

The `config` object's parameters vary for different source types. Read [Source Configuration](#) for information about the available parameters in the `config` object for each supported source type.

Parameters of the `metadataPolicy` Object

authTTLms Body Integer Optional

Length of time to cache the privileges that the user has on the source, in milliseconds. For example, if authTTLms is set to `28800000` (8 hours), Dremio checks the user's privilege status once every 8 hours. Default is `86400000` (24 hours). Minimum is `60000` (1 minute).

Example: 86400000

namesRefreshMs Body Integer Optional

How often to refresh the source, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetRefreshAfterMs Body Integer Optional

How often to refresh the metadata in the source's datasets, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetExpireAfterMs Body Integer Optional

Maximum age to allow for the metadata in the source's datasets, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 10800000

datasetUpdateMode Body String Optional


Approach for Dremio to take for updating the metadata when updating datasets in the source.

• **PREFETCH**: (deprecated) Dremio updates details for all datasets in a source.

• **PREFETCH_QUERIED**: Dremio updates details for previously queried objects in a

source.

Example: PREFETCH_QUERIED

deleteUnavailableDatasets Body  Boolean Optional

If Dremio should remove dataset definitions from the source when the underlying data is unavailable, set to `true` (default). Otherwise, set to `false`.

Example: true

autoPromoteDatasets Body Boolean Optional

If Dremio should automatically format files into tables when a user issues a query, set to `true`. Otherwise, set to `false` (default). Available only for datalake sources, such as Amazon S3 and Hive.

Example: false

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the source and the specific privileges each user should have.

Example: [{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65","permissions":["VIEW_REFLECTION","SELECT"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the source and the specific privileges each role should have.

Example: [{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56","permissions":["ALTER","CREATE_TABLE","DROP","INSERT","DELETE","UPDATE","TRUNCATE","VIEW_REFLECTION","ALTER_REFLECTION","MODIFY","MANAGE_GRANTS","SELECT"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the source.

Example: ebe519ab-20e3-43ff-9b4c-b3ec590c7e65

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the source. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["VIEW_REFLECTION","SELECT"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
  "tag": "T0/Zr1FOY3A=",
  "type": "S3",
  "name": "AWS-S3_testgroup",
  "metadataPolicy": {
    "authTTLms": 86400000,
    "namesRefreshMs": 3600000,
    "datasetRefreshAfterMs": 3600000,
    "datasetExpireAfterMs": 10800000,
    "datasetUpdateMode": "PREFETCH_QUERIED",
    "deleteUnavailableDatasets": true,
    "autoPromoteDatasets": true
  },
  "accelerationGracePeriodMs": 10800000,
  "accelerationRefreshPeriodMs": 3600000,
  "accelerationActivePolicyType": "PERIOD",
  "accelerationNeverExpire": false,
  "accelerationNeverRefresh": false,
  "allowCrossSourceSelection": false,
  "disableMetadataValidityCheck": false,
  "accelerationRefreshOnDataChanges": true,
  "accessControlList": {
    "users": [
      {
        "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
        "permissions": [
```

```

        "ALTER",
        "VIEW_REFLECTION",
        "SELECT"
    ]
}
],
"roles": [
{
    "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
    "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
    ]
}
]
}
}
}'

```

Example Response

```

{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",

```

```

"tag": "RfVMBBMWRvU=",
"type": "S3",
"name": "AWS-S3_testgroup",
"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLMs": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": true
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [

```

```

        "ALTER",
        "VIEW_REFLECTION",
        "SELECT"
    ]
}
],
"roles": [
{
    "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
    "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
    ]
}
]
},
"permissions": [],
"checkTableAuthorizer": true,
"owner": {
    "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
    "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": true
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Deleting a Source

Delete the specified source, including all of the source's contents.

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the source that you want to delete.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Catalog]/(25.x/reference/api/catalog/)[Next

Source Configuration]/(25.x/reference/api/catalog/source/container-source-config)

Source Attributes

Creating a Source

Parameters

Response Status Codes

Retrieving a Source by ID

Parameters

Response Status Codes

Retrieving a Source by Path

Parameters

Response Status Codes

Updating a Source

Parameters

Response Status Codes

Deleting a Source

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/container-space>

Version: 25.x

On this page

Space

Use the Catalog API to retrieve information about spaces and the child objects they contain, as well as to create, update, and delete spaces.

Space Object

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU="
    }
  ]
}
```

```

    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
    "path": [
      "Example-Space",
      "travel_testing"
    ],
    "tag": "i4mn1SmHqVM=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
  },
  {
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
      "Example-Space",
      "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "MODIFY"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",

```

```

    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Space Attributes

entityType String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id String (UUID)

Unique identifier of the space.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name String

Name of the space.

Example: Example-Space

tag String

Unique identifier of the version of the space. Dremio changes the tag whenever the space changes and uses the tag to ensure that PUT requests apply to the most recent version of the space.

Example: zzOQfjY9IU0=

createdAt String

Date and time that the space was created, in UTC format.

Example: 2023-01-12T18:44:43.237Z

children Array of Object

Information about each catalog object in the space.

Example: [{"id": "8da037a1-8e50-422b-9a2b-cafb03f57c71", "path":


```
[{"Example-Space","testfolder"],"tag": "0McuCL4MzBU=", "type":
"CONTAINER","containerType": "FOLDER"}, {"id":
"f32dfe85-32e2-4c31-b2b4-bfd62ab3f473","path":
["Example-Space","travel\_testing"],"tag": "i4mnlSmHqVM=", "type":
"DATASET","datasetType": "VIRTUAL","createdAt": "2023-01-30T17:54:25.547Z"}, {"id":
"7f1c4660-cd7b-40d0-97d1-b8a6f431cbda","path": ["Example-Space","zips"],"tag":
"ITlp8+qyIMQ=", "type": "DATASET","datasetType": "VIRTUAL","createdAt":
"2023-02-08T16:24:25.084Z"}]
```

accessControlList Object

Enterprise only. Information about users and roles with access to the space and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if space-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["MODIFY"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["MODIFY"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the space. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

```
["READ","WRITE","ALTER\_REFLECTION","SELECT","ALTER","VIEW\_REFLECTION","MODI
FY","MANAGE\_GRANTS","CREATE\_TABLE","DROP","EXTERNAL\_QUERY","INSERT","TRU
NCATE","DELETE","UPDATE","EXECUTE","CREATE\_SOURCE","ALL"]
```

owner Object

Information about the space's owner.

Example: {"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43","ownerType": "USER"}

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the catalog object.

Example: 8da037a1-8e50-422b-9a2b-cafb03f57c71

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["Example-Space","testfolder"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the object.

Example: 0McuCL4MzBU=

type String

Type of the catalog object.

Enum: CONTAINER, DATASET, FILE

Example: CONTAINER

containerType String

For catalog entities with the type `CONTAINER`, the type of container.

Enum: FOLDER, FUNCTION

Example: FOLDER

datasetType String

For catalog objects in a space with the type `DATASET`, the datasetType is `VIRTUAL` (spaces cannot contain tables, only views).

Example: VIRTUAL

createdAt String

For catalog objects in a space with the type `DATASET`, date and time that the catalog object was created, in UTC format.

Example: 2023-01-30T17:54:25.547Z

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the space and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the space and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the space's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the space.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise only. Unique identifier of the user or role with access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise only. List of privileges the user or role has on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["MODIFY"]

Creating a Space

Create a new space.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

name Body String

Name of the space. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: Example-Space

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure space-specific access control privileges.

Example: `{"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] }], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] }] }`

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have.

Example: `[{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] }]`

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have.

Example: `[{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] }]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: `["MODIFY"]`

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
```

```
--data-row '{
  "entityType": "space",
  "name": "Example-Space",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "permissions": [],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}
```

```
}  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieving a Space by ID

Retrieve a space and information about its contents by specifying the space's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to retrieve.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in

concert with the `pageToken` query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: `?maxChildren=25`

`pageToken` Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the `maxChildren` query parameter: the first request URL includes `maxChildren` set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified `maxChildren` value, the response includes a `nextPageToken` attribute. Add the `pageToken` query parameter with the `nextPageToken` value to the request URL to retrieve the next page of results. Do not remove or change the `maxChildren` query parameter when you add `pageToken` to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

`?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE`

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McUCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    }
  ],
}
```

```

{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "path": [
    "Example-Space",
    "zips"
  ],
  "tag": "ITlp8+qyIMQ=",
  "type": "DATASET",
  "datasetType": "VIRTUAL",
  "createdAt": "2023-02-08T16:24:25.084Z"
},
{
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Response Status Codes

200 OK

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a Space by Path

Retrieve a space and information about its contents by specifying the space's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the space that you want to retrieve. The path is the name of the space.

Example: Example-Space

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Example-Space' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
        "zips"
      ],
      "tag": "ITlp8+qyIMQ=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-08T16:24:25.084Z"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
```

```

        "MODIFY"
    ]
}
],
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Updating a Space

Update the specified space.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

entityType Body String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id Body String (UUID)

Unique identifier of the space to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name Body String

Name of the space to update.

Example: Example-Space

tag Body String

Unique identifier of the version of the space that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the space.

Example: zzOQfjY9IU0=

accessControlList Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. If you omit the accessControlList object in a PUT request, Dremio removes all existing user and role access settings from the space. To keep existing user and role access settings while making other updates, duplicate the existing accessControlList array in the PUT request.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALL"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["MODIFY"]}]}

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have. If you omit the users object in a PUT request, Dremio removes all existing user access settings from the space. To keep existing user access settings while making other updates, duplicate the existing users array in the PUT request.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have. If you omit the roles object in a PUT request, Dremio removes all existing role access settings from the space. To keep existing role access settings while making other updates, duplicate the existing roles array in the PUT request.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["ALL"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALL"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "PwZ6e/axHUY=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ]
    }
  ]
}
```

```

    ],
    "tag": "0McUCL4MzBU=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
    "path": [
      "Example-Space",
      "travel_testing"
    ],
    "tag": "i4mnlSmHqVM=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
  },
  {
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
      "Example-Space",
      "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER_REFLECTION",
        "MODIFY",
        "ALTER",
        "MANAGE_GRANTS",
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}

```

```
}  
}
```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Deleting a Space

Delete the specified space, including all of the space's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to delete.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0'  
\br/>--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Home](/25.x/reference/api/catalog/container-home)[Next

Folder](/25.x/reference/api/catalog/container-folder)

Space Attributes

Creating a Space

Parameters

Response Status Codes

Retrieving a Space by ID

Parameters

Response Status Codes

Retrieving a Space by Path

Parameters

Response Status Codes

Updating a Space

Parameters

Response Status Codes

Deleting a Space

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/container-folder>

Version: 25.x

On this page

Folder

Use the Catalog API to retrieve information about folders and the child objects they

contain, as well as to create, update, and delete folders.

Folder Object

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
      ],
      "type": "FILE"
    },
    {
      "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "type": "DATASET",
      "datasetType": "PROMOTED"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "roles": [
    {
```

```

    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
      "SELECT"
    ]
  },
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Folder Attributes

entityType String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

id String

Unique identifier of the folder. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

path Array of String

Path of the folder within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target folder itself as

the last item in the array.

Example: ["Samples","samples.dremio.com"]

tag String

Unique identifier of the version of the folder. Dremio changes the tag whenever the folder changes and uses the tag to ensure that PUT requests apply to the most recent version of the folder.

Example: pRmJ0BQ9SFw=

children Array of Object

Information about each catalog object in the folder.

Example: [{"id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv","path": ["Samples","samples.dremio.com","zip_lookup.csv"],"type": "FILE"}, {"id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg","path": ["Samples","samples.dremio.com","NYC-taxi-trips-iceberg"],"type": "CONTAINER","containerType": "FOLDER"}, {"id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d","path": ["Samples","samples.dremio.com","NYC-taxi-trips"],"type": "DATASET","datasetType": "PROMOTED"}]

accessControlList Object

Enterprise-only. Information about users and roles with access to the folder and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if folder-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the folder. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ","WRITE","ALTER_REFLECTION","SELECT","ALTER","VIEW_REFLECTION","MODIFY","MANAGE_GRANTS","CREATE_TABLE","DROP","EXTERNAL_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE_SOURCE","ALL"]

owner Object

Information about the folder's owner.

Example: {"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43","ownerType": "USER"}

Attributes of Objects in the `children` Array

id String

Unique identifier of the catalog object. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: `dremio:/Samples/samples.dremio.com/zip_lookup.csv`

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: `["Samples","samples.dremio.com","zip_lookup.csv"]`

type String

Type of the catalog object. If the object is saved within a space (including the home space), valid types are ``CONTAINER`` and ``DATASET``. If the object is saved within a source, valid types are ``CONTAINER``, ``FILE``, and ``DATASET``.

Example: `CONTAINER`

containerType String

For catalog objects with the type ``CONTAINER``, the containerType is ``FOLDER``.

Example: `FOLDER`

datasetType String

For catalog objects with the type ``DATASET``, the type of dataset. For tables, the datasetType is ``PROMOTED``. For views, the datasetType is ``VIRTUAL``.

Enum: `PROMOTED`, `VIRTUAL`

Example: `VIRTUAL`

createdAt String

Date and time that the catalog object was created, in UTC format. The createdAt attribute is included only for ``DATASET`` catalog objects that are saved in folders within spaces, not within sources.

Example: `2023-01-30T17:54:25.547Z`

Attributes of the ``accessControlList`` Object

users String

Enterprise-only. List of users with access to the folder and the specific privileges each user has.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}]`

roles String

Enterprise-only. List of roles whose members have access to the folder and the specific privileges each role has.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions":["SELECT"]}]`

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the folder's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the folder.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise-only. Unique identifier of the user or role with access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise-only. List of privileges the user or role has on the folder. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Creating a Folder

Create a new folder within a space.

note

The Catalog API cannot create new folders within sources.

Method and URL

POST /api/v3/catalog

Parameters

entityType Body String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

path Body Array of String

Path of the location where the folder should be created within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the target folder itself as the last item in the array. The name of the folder cannot include the following special characters: `/`, `:`, `[`, or ` `.

Example: ["Example-Space","First-Folder","New-Folder"]

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure folder-specific access control privileges.

Example: {"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["SELECT","ALTER"]}], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]}

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["SELECT","ALTER"]}]

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more

information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["SELECT","ALTER"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {

```

```

    "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
    "permissions": [
      "SELECT",
      "ALTER"
    ]
  },
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Retrieving a Folder by ID

Retrieve a folder and information about its contents by specifying the folder's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to retrieve. If the ID is a text path, use URL

encoding to replace any special characters with their UTF-8-equivalent characters, such as `%3A` for a colon; `%2F` for a forward slash; and `%20` for a space. For example, if the ID value is `dremio:/Samples/samples.dremio.com/Dremio University`, the URI-encoded ID is `dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for folders in filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/d4c2a8ba-a972-4db4-8deb-67e1ade684d1' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
      ],
      "type": "FILE"
    },
    {
      "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "type": "DATASET",
      "datasetType": "PROMOTED"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
```

```

        "SELECT"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a Folder by Path

Retrieve a folder and information about its contents by specifying the folder's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the folder that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Samples/samples.dremio.com

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdkyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
```

```

        "samples.dremio.com",
        "zip_lookup.csv"
    ],
    "type": "FILE"
},
{
    "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
    "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
    "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
    ],
    "type": "DATASET",
    "datasetType": "PROMOTED"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "SELECT",
                "ALTER"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "SELECT"
            ]
        }
    ]
},
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found

Updating a Folder

Update the specified folder.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

entityType Body String

Type of the catalog object. For folders, the entityType is ``folder``.

Example: folder

id Body String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

path Body Array of String

Path of the location where the folder is saved within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the target folder itself as the last item in the array.

Example: ["Example-Space", "First-Folder", "New-Folder"]

tag Body String

Unique identifier of the version of the folder that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the folder.

Example: R7COubQq8KE=

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. If you omit the `accessControlList` object in a PUT request, Dremio removes all existing user and role access settings from the folder. To keep existing user and role access settings while making other updates, duplicate the existing `accessControlList` array in the PUT request.

Example: `{"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"] }], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"] }]}`

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have. If you omit the `users` object in a PUT request, Dremio removes all existing user access settings from the folder. To keep existing user access settings while making other updates, duplicate the existing `users` array in the PUT request.

Example: `[{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"] }]`

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have. If you omit the `roles` object in a PUT request, Dremio removes all existing role access settings from the folder. To keep existing role access settings while making other updates, duplicate the existing `roles` array in the PUT request.

Example: `[{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"] }]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["ALL"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALL"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "vnRnYLLpCFU=",
  "children": [
    {
      "id": "d60f9258-e55a-4fc3-97b3-58c6720a70fc",
      "path": [
        "Example-Space",
        "First-Folder",
        "New-Folder",

```



```

        "NYC-trips-weather"
    ],
    "tag": "IHXU70xs80c=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "acba8595-bfcf-4126-887c-d2a19b5afb1d",
    "path": [
        "Example-Space",
        "First-Folder",
        "New-Folder",
        "short-distance-trips"
    ],
    "tag": "KYs/Qywlok8=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-09T19:09:58.789Z"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "ALTER_REFLECTION",
                "ALTER",
                "MANAGE_GRANTS",
                "VIEW_REFLECTION",
                "SELECT"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "SELECT"
            ]
        }
    ]
},
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Deleting a Folder

Delete the specified folder, including all of the folder's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to delete. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: 598697c2-8be0-4050-9731-53563977a17d

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/598697c2-8be0-4050-9731-53563977a17d' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

- 204 No Content
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Was this page helpful?

[Previous

Space](/25.x/reference/api/catalog/container-space)[Next

File](/25.x/reference/api/catalog/file)

Folder Attributes

Creating a Folder

Parameters

Response Status Codes

Retrieving a Folder by ID

Parameters

Response Status Codes

Retrieving a Folder by Path

Parameters

Response Status Codes

Updating a Folder

Parameters

Response Status Codes

Deleting a Folder

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/container-home>

Version: 25.x

On this page

Home

Use the Catalog API to retrieve information about the home space and the child objects it contains.

Home Object

```
{
  "entityType": "home",
  "id": "87049e43-8564-4ee7-8bb6-5bdaf5bd0959",
  "name": "@user@dremio.com",
```

```

"tag": "8S9cTZ5IsWo=",
"children": [
  {
    "id": "1e16c0e5-c890-4f87-b1a6-ac9325aafa2c",
    "path": [
      "@user@dremio.com",
      "Business"
    ],
    "tag": "KgFBPW3+Cyc=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "a59815d0-3c21-41ad-b9bc-2ba105251fa6",
    "path": [
      "@user@dremio.com",
      "meeting_rooms_lookup"
    ],
    "tag": "0aQT64frevc=",
    "type": "DATASET",
    "datasetType": "PROMOTED",
    "createdAt": "1970-01-01T00:00:00.000Z"
  },
  {
    "id": "37401663-8666-4e00-bc03-668abb43ccd7",
    "path": [
      "@user@dremio.com",
      "NYC-trips-quarterly"
    ],
    "tag": "+H5TpLYoosY=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-07T21:26:14.385Z"
  }
]
}

```

Home Attributes

entityType String

Type of the catalog object. For the home space, the entityType is `home`.

Example: home

id String (UUID)

Unique identifier of the home space.

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

name String

Name of the home space. Automatically generated based on the username.

Example: @user@dremio.com

tag String

Unique identifier of the version of the home space. Dremio uses tags to ensure that PUT requests apply to the most recent version of the resource being updated. However, home spaces cannot be changed, so the tag is listed in the home space object but not used.

Example: 8S9cTZ5lsWo=

children Array of Object

Information about each catalog object in the home space.

Example: [{"id":"1e16c0e5-c890-4f87-b1a6-ac9325aafa2c","path":["@dremio","Business"],"tag":"KgFBPW3+Cyc=","type":"CONTAINER","containerType":"FOLDER"}, {"id":"a59815d0-3c21-41ad-b9bc-2ba105251fa6","path":["@dremio","meeting_rooms_lookup"],"tag":"OaQT64frevc=","type":"DATASET","datasetType":"PROMOTED","createdAt":"1970-01-01T00:00:00.000Z"}, {"id":"37401663-8666-4e00-bc03-668abb43ccd7","path":["@dremio","NYC-trips-quarterly"],"tag":"+H5TpLYoosY=","type":"DATASET","datasetType":"VIRTUAL","createdAt":"2023-02-07T21:26:14.385Z"}]

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the catalog object.

Example: 1e16c0e5-c890-4f87-b1a6-ac9325aafa2c

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the home space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["@user@dremio.com","Business"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the object.

Example: KgFBPW3+Cyc=

type String

Type of the catalog object.

Enum: CONTAINER, DATASET

Example: CONTAINER

containerType String

For catalog entities with the type `CONTAINER`, the type of container.

Enum: FOLDER, FUNCTION

Example: FOLDER

datasetType String

For catalog entities with the type `DATASET`, the type of dataset. For tables, the datasetType is `PROMOTED`. For views, the datasetType is `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: PROMOTED

createdAt String

For catalog entities with the type `DATASET`, date and time that the catalog object was created, in UTC format.

Example: 2023-02-07T21:26:14.385Z

Retrieving the Home Space by ID

Retrieve information about the home space and its contents by specifying the home space's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Body String (UUID)

Unique identifier of the home space that you want to retrieve.

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the home space has more child objects than the specified maxChildren value, the response includes a

nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/63505c60-bc86-42aa-a622-24e5f22ce50b' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "home",
  "id": "87049e43-8564-4ee7-8bb6-5bdaf5bd0959",
  "name": "@user@dremio.com",
  "tag": "8S9cTZ5IsWo=",
  "children": [
    {
      "id": "1e16c0e5-c890-4f87-b1a6-ac9325aafa2c",
      "path": [
        "@user@dremio.com",
        "Business"
      ],
      "tag": "KgFBPW3+Cyc=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "a59815d0-3c21-41ad-b9bc-2ba105251fa6",
      "path": [
        "@user@dremio.com",
        "meeting_rooms_lookup"
      ],
      "tag": "0aQT64frevc=",
      "type": "DATASET",
      "datasetType": "PROMOTED",
      "createdAt": "1970-01-01T00:00:00.000Z"
    },
    {
      "id": "37401663-8666-4e00-bc03-668abb43ccd7",
      "path": [
        "@user@dremio.com",
        "NYC-trips-quarterly"
      ],
      "tag": "+H5TpLYoosY=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-07T21:26:14.385Z"
    }
  ]
}
```

```
]
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving the Home Space by Path

Retrieve information about the home space and its contents by specifying the home space's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the home space whose information you want to retrieve. The home space path is the username, preceded with the `@` symbol.

Example: @user@dremio.com

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the `pageToken` query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: `?maxChildren=25`

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the `maxChildren` query parameter: the first request URL includes `maxChildren` set to the maximum number of child objects to include in each page of results. If the home space has more child objects than the specified `maxChildren` value, the response includes a `nextPageToken` attribute. Add the `pageToken` query parameter with the `nextPageToken` value to the request URL to retrieve the next page of results. Do not remove or change the `maxChildren` query parameter when you add `pageToken` to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/@user@dremio.com' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "home",
  "id": "87049e43-8564-4ee7-8bb6-5bdaf5bd0959",
  "name": "@user@dremio.com",
  "tag": "8S9cTZ5IsWo=",
  "children": [
    {
      "id": "1e16c0e5-c890-4f87-b1a6-ac9325aafa2c",
      "path": [
        "@user@dremio.com",
        "Business"
      ],
      "tag": "KgFBPW3+Cyc=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "a59815d0-3c21-41ad-b9bc-2ba105251fa6",
      "path": [
        "@user@dremio.com",
        "meeting_rooms_lookup"
      ],
      "tag": "0aQT64frevc=",
      "type": "DATASET",
      "datasetType": "PROMOTED",
      "createdAt": "1970-01-01T00:00:00.000Z"
    },
    {
      "id": "37401663-8666-4e00-bc03-668abb43ccd7",
      "path": [
        "@user@dremio.com",
        "NYC-trips-quarterly"
      ],
      "tag": "+H5TpLYoosY=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-07T21:26:14.385Z"
    }
  ]
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Source Configuration](/25.x/reference/api/catalog/source/container-source-config)[Next
Space](/25.x/reference/api/catalog/container-space)

Home Attributes

Retrieving the Home Space by ID

Parameters

Response Status Codes

Retrieving the Home Space by Path

Parameters

Response Status Codes

Source:
[https://docs.dremio.com/25.x/reference/api/cat
alog/file](https://docs.dremio.com/25.x/reference/api/catalog/file)

Version: 25.x

On this page

File

Use the Catalog API to retrieve information about [formatting data to a table](#).

File Object

```
{
  "entityType": "file",
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
  "path": [
    "Samples",
    "samples.dremio.com",
    "SF weather 2018-2019.csv"
  ]
}
```

```
}
```

File Attributes

entityType String

Type of the catalog object. For files, the entityType is `file`.

Example: file

id String

Unique identifier of the file. For files, the ID is the text path of the file within Dremio.

Example: dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv

path Array of String

Path of the file within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target file itself as the last item in the array.

Example: ["Samples","samples.dremio.com","SF weather 2018-2019.csv"]

Retrieving a File by Path

Retrieve information about a file by specifying its path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the file that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, replace colons with `%3A` and replace spaces with `%20`.

Example: Samples/samples.dremio.com/SF%20weather%202018-2019.csv

Example Request

```
curl -X GET \
'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/SF%20weather%2018-2019.csv' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "file",
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
  "path": [
    "Samples",
    "samples.dremio.com",
    "SF weather 2018-2019.csv"
  ]
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Was this page helpful?

[Previous

Folder](/25.x/reference/api/catalog/container-folder)[Next

Table](/25.x/reference/api/catalog/table)

File Attributes

Retrieving a File by Path

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/table>

Version: 25.x

On this page

Table

Use the Catalog API to retrieve tables, format files and folders as tables, update and

refresh tables, and revert tables to files and folders.

Table Object

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "SCHEDULE",
    "refreshPeriodMs": 3600000,
    "gracePeriodMs": 10800000,
    "refreshSchedule": "0 0 8 * * ?",
    "method": "FULL",
    "neverExpire": false,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
```

```

        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
    ]
}
],
"roles": [
{
    "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
    "permissions": [
        "SELECT",
        "ALTER"
    ]
}
]
},
"permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
],
"owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
},
"fields": [
{
    "name": "_id",
    "type": {
        "name": "VARCHAR"
    }
},
{
    "name": "name",
    "type": {
        "name": "VARCHAR"
    }
}
],
{

```

```

    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "attributes",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "Parking",
          "type": {
            "name": "STRUCT",
            "subSchema": [
              {
                "name": "garage",
                "type": {
                  "name": "BOOLEAN"
                }
              }
            ]
          }
        },
        {
          "name": "street",

```

```

        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "lot",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "valet",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  },
  {
    "name": "Accepts Credit Cards",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Wheelchair Accessible",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Price Range",
    "type": {
      "name": "BIGINT"
    }
  }
]
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Table Attributes

entityType String

Type of the catalog object. For tables, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the table.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

type String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example: PHYSICAL_DATASET

path Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples","samples.dremio.com","Dremio University","restaurant_reviews.parquet"]

createdAt String

Date and time that the table was created, in UTC format.

Example: 2024-01-13T19:52:01.894Z

tag String (UUID)

Unique identifier of the version of the table. Dremio changes the tag whenever the table changes and uses the tag to ensure that PUT requests apply to the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

accelerationRefreshPolicy String

Attributes that define the acceleration refresh policy for the table.

isMetadataExpired Boolean

If true, the metadata of the table needs to be refreshed. To refresh it, run the ALTER TABLE command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the table.

If NULL, metadata has never yet been collected for the table.

lastMetadataRefreshAt String

Date and time that the table metadata was last refreshed. In UTC format. If NULL, the metadata has never yet been refreshed.

Example: 2024-01-31T09:50:01.012Z

format Object

Table format attributes.

accessControlList Object

Enterprise only. Information about users and roles with access to the table and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if table-specific access control privileges are not set.

Example: `{"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ _GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}`

permissions Array of String

Enterprise-only. List of the privileges that you have on the table. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

`["READ", "WRITE", "ALTER\ _REFLECTION", "SELECT", "ALTER", "VIEW\ _REFLECTION", "MODIFY", "MANAGE\ _GRANTS", "CREATE\ _TABLE", "DROP", "EXTERNAL\ _QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE\ _SOURCE", "ALL"]`

owner String

Information about the table's owner.

fields Object

Attributes that represent the table schema.

approximateStatisticsAllowed Boolean

If true, `COUNT DISTINCT` queries run on the table return approximate results. Otherwise, false.

Example: `{"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8", "ownerType": "USER"}`

Attributes of the `accelerationRefreshPolicy` Object

activePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `neverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is

defined by refreshPeriodMs.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by refreshSchedule.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency.

refreshPeriodMs Integer

Refresh period for the data in all Reflections for the table, in milliseconds.

Example: 3600000

refreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed.

Field	Allowed Values	Allowed Special Characters
-------	----------------	----------------------------

---	---	---
-----	-----	-----

Second	0 N/A	
--------	---------	--

Minute	0-59 N/A	
--------	------------	--

Hour	0-23 N/A	
------	------------	--

Day of month	N/A * ?	
--------------	------------	--

Month	N/A * ?	
-------	------------	--

Days of week	1-7 or SUN-SAT , - * ?	
--------------	---------------------------	--

Special Character	Description
-------------------	-------------

---	---
-----	-----

*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
----	--

?	Equivalent to *.
---	-------------------

,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
---	---

-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.
---	--

Examples:

``0 0 0 * * ?`` : Refreshes every day at midnight.

``0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

``0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Integer

Maximum age allowed for Reflection data used to accelerate queries, in milliseconds.

Example: 10800000

method String

Approach used for refreshing the data in Reflections defined on tables that are not in the Apache Iceberg format. For more information, read [Refreshing Reflections](#).

Enum: FULL, INCREMENTAL

Example: FULL

refreshField String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Boolean

If the Reflection never expires, the value is `true`. Otherwise, the value is `false`.

Example: false

neverRefresh Boolean

If the Reflection never refreshes, the value is `true`. Otherwise, the value is `false`.

Example: false

sourceRefreshOnDataChanges Boolean

If the table's source is configured so that Reflections on tables in Iceberg format in the source will refresh when new snapshots are created after an update, `true`. Otherwise, `false`.

Attributes of the `format` Object

type String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

Example: Parquet

name String

Table name. Dremio automatically duplicates the name of the origin file or folder to populate this value. The name of the origin file or folder cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: restaurant_reviews.parquet

fullPath Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples", "samples.dremio.com", "Dremio University",

"restaurant_reviews.parquet"]

ctime Integer

Not used. Has the value `0`.

Example: 0

isFolder Boolean

If the value is `true`, the table was created from a folder. If the value is `false`, the table was created from a file.

Example: false

location String

Location, expressed as a string, where the table's metadata is stored within a Dremio source or space.

Example: /samples.dremio.com/Dremio University/restaurant_reviews.parquet

ignoreOtherFileFormats Boolean

If true, Dremio ignores all non-Parquet files in the related folder structure, and the promoted table works as if only Parquet files are in the folder structure. Otherwise, false. Included only for Parquet folders.

Example: false

metaStoreType String

Not used. Has the value `HDFS`.

Example: HDFS

parquetDataFormat Object

Information about data format for Parquet tables.

dataFormatTypeList Array of String

List of data format types in the table. Included only for Iceberg tables, and `PARQUET` is the only valid value.

Example: ["PARQUET"]

sheetName String

For tables created from files that contain multiple sheets, the name of the sheet used to create the table.

Example: location_1

extractHeader Boolean

For tables created from files, the value is `true` if Dremio extracted the table's column names from the first line of the file. Otherwise, the value is `false`.

Example: false

hasMergedCells Boolean

For tables created from files, the value is `true` if Dremio expanded merged cells in the file when creating the table. Otherwise, the value is `false`.

Example: true

fieldDelimiter String

Character used to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character.

quote String

Character used for quotation marks in the table. May be `"` for a double quote (default), `'` for a single quote, or a custom character.

comment String

Character used to indicate comments in the table. May be `#` for a number sign (default) or a custom character.

escape String

Character used to indicate an escape in the table. May be `"` for a double quote (default), ``` for a back quote, `\` for a backward slash, or a custom character.

lineDelimiter String

Character used to indicate separate lines in the table. May be `\r\n` for a carriage return and a new line (default), `\n` for a new line, or a custom character.

skipFirstLine Boolean

If Dremio skipped the first line in the file or folder when creating the table, the value is `true`. Otherwise, the value is `false`.

Example: false

autoGenerateColumnNames Boolean

If Dremio used the existing column names in the file or folder for the table columns, the value is `true`. Otherwise, the value is `false`.

Example: true

trimHeader Boolean

If Dremio trimmed column names to a specific number of characters when creating the table, the value is `true`. Otherwise, the value is `false`.

Example: true

autoCorrectCorruptDates Boolean

If Dremio automatically corrects corrupted date fields in the table, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]]

roles Array of Object

Enterprise only. List of roles whose members have access to the table and the specific privileges each role has.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT", "ALTER"]}]]

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the table's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the table.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the table field.

Example: review_count

type Object

Information about the table field.

Attributes of the `type` Object

name String

Name of the table field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: BIGINT

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 0

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema appears only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: Parking

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Attributes of the `parquetDataFormat` Object

type String

Type of data in the table. Within the `parquetDataFormat` object, the only valid type is ``Parquet``.

Example: Parquet

ctime Integer

Not used. Has the value ``0``.

Example: 0

isFolder Boolean

If the value is ``true``, the table was created from a folder. If the value is ``false``, the table was created from a file.

Example: true

autoCorrectCorruptDates Boolean

If the value is ``true``, Dremio automatically corrects corrupted date fields in the table. Otherwise, the value is ``false``.

Example: true

Formatting a File or Folder as a Table

Format a file or folder as a table so that you can query the data in Dremio.

note

To format a folder, all files in the folder must be the same format.

Method and URL

```
POST /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the file or folder you want to format. The ID can be a UUID or a text path. If the ID is a text path, use URL encoding to replace special characters with their UTF-8-equivalent characters: ``%3A`` for a colon; ``%2F`` for a forward slash, and ``%20`` for a space. For example, if the ID value is ``dremio:/Samples/samples.dremio.com/Dremio University``, the URI-encoded ID is ``dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University``.

Example: c590ed7f-7142-4e1f-ba7d-94173afdc9a3

entityType Body String

Type of the catalog object. To format a file or folder as a table, the `entityType` is

``dataset``.

`path` Body Array of String

Path of the file or folder you want to format, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the file or folder itself as the last item in the array. Get the path from the file or folder's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

`type` Body String

Type of dataset. For tables, the type is ``PHYSICAL_DATASET``.

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

`format` Body String

Parameters that describe how to format the file or folder.

[accessControlList](#) Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

Parameters of the ``accelerationRefreshPolicy`` Object

`activePolicyType` Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the `neverRefresh` parameter must be set to ``false``.

The possible values are:

``NEVER``: The Reflections are never refreshed.

``PERIOD``: Default. The Reflections are refreshed at the end of every period that is defined by `refreshPeriodMs`.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in

Iceberg format.

refreshPeriodMs Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set activePolicyType to `PERIOD`. The default setting is `3600000` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

refreshSchedule Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set activePolicyType to `SCHEDULE`. The default refreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0 N/A	
Minute	0-59 N/A	
Hour	0-23 N/A	
Day of month	N/A * ?	
Month	N/A * ?	
Days of week	1-7 or SUN-SAT , - * ?	

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
?	Equivalent to *.
,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the file or folder. All files in the folder must be the same format.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

ignoreOtherFileFormats Body Boolean Optional

If Dremio should ignore all non-Parquet files in the related folder structure so that the promoted table works as if only Parquet files are in the folder structure, set to `true`. Otherwise, set to `false` (default). Optional for Parquet folders.

Example: false

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the file or folder when creating the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for files or folders of the Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to ``true``. Otherwise, set to ``false`` (default). Optional for files or folders of the Excel and Text types.

Example: "

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the file when creating the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for files or folders of the Excel type.

Example: true

sheetName Body String Optional

For tables created from Excel files that contain multiple sheets, the name of the sheet to use to create the table. Default is the first sheet in the file (for files that contain multiple sheets).

Example: location_1

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be ```,`` for a comma (default), ``\t`` for a tab, ``|`` for a pipe, or a custom character. Optional for files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be ``"``` for a double quote (default), ``'`` for a single quote, or a custom character. Optional for files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments in the table. May be ``#`` for a number sign (default) or a custom character. Optional for files or folders of the Text type.

escape Body String Optional

Character used to indicate an escape in the table. May be ``"``` for a double quote (default), ``\`` for a back quote, ``\`` for a backward slash, or a custom character. Optional for files or folders of the Text type.

lineDelimiter Body String Optional

Character used to indicate separate lines in the table. May be ``\r\n`` for a carriage return and a new line (default), ``\n`` for a new line, or a custom character. Optional for files or folders of the Text type.

autoGenerateColumnNames Body Boolean Optional

If Dremio should use the existing column names in the file or folder for the table columns, set to `true` (default). Otherwise, set to `false`. Optional for files or folders of the Text type.

Example: true

trimHeader Body Boolean Optional

If Dremio should trim column names to a specific number of characters when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT","ALTER"]},{ "id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT","ALTER","MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
```

```
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Samples",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Parquet"
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ]
}
```

```

],
"createdAt": "2024-01-13T19:52:01.894Z",
"tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
"accelerationRefreshPolicy": {
  "activePolicyType": "PERIOD",
  "refreshPeriodMs": 3600000,
  "refreshSchedule": "0 56 18 * * *",
  "gracePeriodMs": 259200000,
  "method": "FULL",
  "neverExpire": true,
  "neverRefresh": false,
  "sourceRefreshOnDataChanges": false
},
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"format": {
  "type": "Parquet",
  "name": "restaurant_reviews.parquet",
  "fullPath": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "ctime": 0,
  "isFolder": false,
  "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
  "ignoreOtherFileFormats": "false",
  "autoCorrectCorruptDates": true
},
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
}

```



```

    ]
  }
]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",

```

```

    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "attributes",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "Parking",
          "type": {
            "name": "STRUCT",
            "subSchema": [
              {
                "name": "garage",
                "type": {
                  "name": "BOOLEAN"
                }
              },
              {
                "name": "street",
                "type": {
                  "name": "BOOLEAN"
                }
              },
              {
                "name": "lot",
                "type": {
                  "name": "BOOLEAN"
                }
              },
              {
                "name": "valet",
                "type": {
                  "name": "BOOLEAN"
                }
              }
            ]
          }
        }
      ],
    },
    {
      "name": "Accepts Credit Cards",
      "type": {
        "name": "BOOLEAN"
      }
    },
    {
      "name": "Wheelchair Accessible",
      "type": {
        "name": "BOOLEAN"
      }
    }
  },
  {

```

```

        "name": "Price Range",
        "type": {
            "name": "BIGINT"
        }
    }
]
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],
"approximateStatisticsAllowed": false
}

```

Example Request for Excel format type

```

curl -X POST 'https://{hostname}/api/v3/catalog/dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University%2Foracle-departments.xlsx' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
    "entityType": "dataset",
    "path": [
        "Samples",
        "samples.dremio.com",
        "Dremio University",
        "oracle-departments.xlsx"
    ],
    "type": "PHYSICAL_DATASET",
    "format": {
        "type": "Excel",
        "extractHeader": true,
        "hasMergedCells": true,
        "sheetName": "Sheet1"
    }
}'

```

Example Request for Text format type

```

curl -X POST 'https://{hostname}/api/v3/catalog/6ba3bd6e-fd27-4572-a535-77e1548283b3' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
    "entityType": "dataset",
    "path": [
        "Samples",

```

```

    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Text",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": false
  }
}'

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a Table by ID

Retrieve a table by specifying the table's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to retrieve.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
```

```

"users": [
  {
    "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
    "permissions": [
      "SELECT",
      "ALTER"
    ]
  },
  {
    "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "permissions": [
      "SELECT",
      "ALTER",
      "MANAGE_GRANTS"
    ]
  }
],
"roles": [
  {
    "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
    "permissions": [
      "SELECT",
      "ALTER"
    ]
  }
],
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  }
]

```

```

    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "attributes",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "Parking",
          "type": {
            "name": "STRUCT",
            "subSchema": [
              {
                "name": "garage",
                "type": {
                  "name": "BOOLEAN"
                }
              }
            ],
            "name": "street",
            "type": {
              "name": "BOOLEAN"
            }
          }
        },
        {
          "name": "lot",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  }

```

```

        },
        {
            "name": "valet",
            "type": {
                "name": "BOOLEAN"
            }
        }
    ]
}
},
{
    "name": "Accepts Credit Cards",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Wheelchair Accessible",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Price Range",
    "type": {
        "name": "BIGINT"
    }
}
]
}
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Retrieving a Table by Path

Retrieve a table by specifying the table's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Table's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" table in the "samples.dremio.com" folder within the source "Samples," the path is `Samples/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example:

Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
```

```

    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
},

```

```

"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  }
],

```

```

{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "street",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "lot",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "valet",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ]
        }
      },
      {
        "name": "Accepts Credit Cards",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "Wheelchair Accessible",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "Price Range",
        "type": {
          "name": "BIGINT"
        }
      }
    ]
  }
}

```

```

    }
  ]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Updating a Table

Update the specified table in Dremio.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

entityType Body String

Type of the catalog object. For tables, the entityType is `dataset`.

id Body String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

path Body Array of String

Path of the table that you want to update, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array. Get the path from the table's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

tag Body String (UUID) Optional

Unique identifier of the version of the table that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the table. If you do not provide a tag, Dremio automatically updates the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

type Body String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example:

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

[format](#) Body String

Parameters that describe the table's format.

[accessControlList](#) Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

Parameters of the `accelerationRefreshPolicy` Object

activePolicyType Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the neverRefresh parameter must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by refreshPeriodMs.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in Iceberg format.

`refreshPeriodMs` Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set `activePolicyType` to ``PERIOD``. The default setting is ``3600000`` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

`refreshSchedule` Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set `activePolicyType` to ``SCHEDULE``. The default `refreshSchedule` setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
-------	----------------	----------------------------

---	---	---
-----	-----	-----

Second	0	N/A
--------	---	-----

Minute	0-59	N/A
--------	------	-----

Hour	0-23	N/A
------	------	-----

Day of month	N/A	* ?
--------------	-----	------

Month	N/A	* ?
-------	-----	------

Days of week	1-7 or SUN-SAT	, - * ?
--------------	----------------	----------

Special Character	Description
-------------------	-------------

---	---
-----	-----

*	Used to specify all values for a field. For <code>`Day of month`</code> , specifies every day of the month. For <code>`Month`</code> , specifies every month. For <code>`Days of week`</code> , specifies every day of the week.
----	--

?	Equivalent to *.
---	-------------------

,	Used to specify two or more days in the <code>`Days of week`</code> field. For example, <code>`MON,WED,FRI`</code> .
---	--

-	Used to specify ranges in the <code>`Days of week`</code> field. For example, <code>`1-3`</code> is equivalent to <code>`Sunday, Monday, and Tuesday`</code> .
---	--

Examples:

``0 0 0 * * ?`` : Refreshes every day at midnight.

``0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

``0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

`gracePeriodMs` Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to ``true``. Otherwise, set to ``false`` (default). Optional for tables created from files or folders of the Excel and Text types.

Example: true

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for tables created from files or folders of the Excel type.

Example: true

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be ``,`` for a comma (default), ``\t`` for a tab, ``|`` for a pipe, or a custom character. Optional for tables created from files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be ``\"`` for a double quote (default), ``'`` for a single quote, or a custom character. Optional for tables created from files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments for the table. May be ``#`` for a number sign (default) or a custom character. Optional for tables created from files or folders of the Text type.

escape Body String Optional

Character to use to indicate an escape for the table. May be ``\"`` for a double quote (default), ``\`` for a back quote, ``\\`` for a backward slash, or a custom character. Optional for tables created from files or folders of the Text type.

lineDelimiter Body String Optional

Character to use to indicate separate lines for the table. May be ``\r\n`` for a carriage return and a new line (default), ``\n`` for a new line, or a custom character. Optional for tables created from files or folders of the Text type.

Example:

autoGenerateColumnNames Body Boolean Optional

If Dremio should use the existing column names for the table columns, set to ``true`` (default). Otherwise, set to ``false``. Optional for tables created from files or folders of

the Text type.

Example: true

`trimHeader` Body Boolean Optional

If Dremio should trim column names to a specific number of characters when updating the table, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

`users` Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example:

```
[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]
```

`roles` Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example:

```
[{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]
```

Parameters of Objects in the `users` and `roles` Arrays

`id` Body String

Enterprise only. Unique identifier of the user or role that should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

`permissions` Body Array of String

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/dba1e4fe-6351-44d2-a3e0-7aa20e782bf3' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "dba1e4fe-6351-44d2-a3e0-7aa20e782bf3",
  "path": [
```

```

    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Text",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  }
}'

```

Example Response

```

{
  "entityType": "dataset",
  "id": "dba1e4fe-6351-44d2-a3e0-7aa20e782bf3",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "createdAt": "2024-01-23T21:26:59.568Z",
  "tag": "fc1707df-35a1-45c1-87d7-5f66fb11a729",
  "format": {
    "type": "Text",
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/airbnb_listings.csv",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  },
  "accessControlList": {},
  "owner": {
    "ownerId": "c590ed7f-7142-4e1f-ba7d-94173afdc9a3",
    "ownerType": "USER"
  }
}

```

```

},
"fields": [
  {
    "name": "id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "listing_url",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "scrape_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "last_scraped",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "summary",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "reviews_per_month",
    "type": {
      "name": "VARCHAR"
    }
  }
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 500 Internal Server Error

Refreshing the Reflections on a Table

Refresh the Reflections associated with the specified table.

note

Refreshing a table's Reflections does not refresh its metadata. Read [Refreshing Metadata](#) to learn how to refresh table metadata. Read [Refreshing Reflections](#) for more information about refreshing Reflections.

Method and URL

```
POST /api/v3/catalog/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

- 204 No Content
- 400 Bad Request
- 401 Unauthorized

403 Forbidden

404 Not Found

Reverting a Table to a File or Folder

Revert a table in a source to change the data in the table back to its original format, file or folder. For more information, read [Formatting Data to a Table](#) and [Removing Formatting on Data](#)

note

If a table is saved in your home space, the revert request will delete the table entirely. The revert endpoint only changes a table back to a file or folder if the table is saved in a source.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to revert to a file or folder.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

File](/25.x/reference/api/catalog/file)[Next

User-Defined Function](/25.x/reference/api/catalog/user-defined-function)

Table Attributes

Formatting a File or Folder as a Table

Parameters

Response Status Codes

Retrieving a Table by ID

Parameters

Response Status Codes

Retrieving a Table by Path

Parameters

Response Status Codes

Updating a Table

Parameters

Response Status Codes

Refreshing the Reflections on a Table

Parameters

Response Status Codes

Reverting a Table to a File or Folder

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/user-defined-function>

Version: 25.x

On this page

User-Defined Function

Use the Catalog API to retrieve information about user-defined functions (UDFs), as well

as to create, update, and delete UDFs.

User-Defined Function Object

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\"
WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\"
DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  },
  "permissions": [],
  "owner": {
    "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
    "ownerType": "USER"
  }
}
```

User-Defined Function Attributes

entityType String

Type of the catalog object. For user-defined functions, the entityType is `function`.

id String (UUID)

Unique identifier of the user-defined function.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

path Array of String

Path of the user-defined function within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the function itself as the last item in the array.

Example: ["team_folder","test_subfolder","filter_domain_orderdates"]

tag String

Unique identifier of the version of the user-defined function. Dremio changes the tag whenever the function changes and uses the tag to ensure that PUT requests apply to the most recent version of the function.

Example: qBWpD7x6+Ws=

createdAt String

Date and time at which the user-defined function was created, in UTC format.

Example: 2024-08-01T20:20:38.547Z

lastModified String

Date and time at which the user-defined function was last modified, in UTC format.

Example: 2024-08-01T20:20:38.547Z

isScalar Boolean

If the user-defined function is a scalar function, `true`. If the user-defined function is a tabular function, `false`.

Example: false

functionArgList String

The user-defined function's arguments and their data types. If the function includes multiple arguments, the arguments are separated with a comma.

note

In response objects, the functionArgList value may contain aliases for data types, such as `CHARACTER VARYING` (an alias for `VARCHAR`).

Example: "domain" CHARACTER VARYING, "orderdate" DATE

functionBody String

The statement that the user-defined function executes.

Example: SELECT "name", "email", "order_date" FROM "customer_data" WHERE LOWER("email") LIKE '%' || LOWER(domain) AND "order_date" >= orderdate

returnType String

The data_type of the result that the function returns (for scalar functions) or of each column that the function returns, separated by commas (for tabular functions).

Example: "name" CHARACTER VARYING, "email" CHARACTER VARYING, "order_date" DATE

accessControlList Object

Enterprise only. Information about users and roles with access to the user-defined function and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if function-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALTER","EXECUTE"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["ALTER","EXECUTE"]}]}]

permissions Array of String

Enterprise-only. List of the privileges that you have on the user-defined function. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read User-Defined Function (UDF) Privileges.

owner Object

Information about the user-defined function's owner.

Example: {"ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349","ownerType": "USER"}

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the user-defined function and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER","EXECUTE"]}]]

roles Array of Object

Enterprise only. List of roles whose members have access to the user-defined function and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER","EXECUTE"]}]]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the user-defined function's owner.

Example: 4740ab48-39c6-434c-9086-8f6e52e65349

ownerType String

Type of owner of the user-defined function.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the user-defined function.

Example: 4740ab48-39c6-434c-9086-8f6e52e65349

permissions Array of String

Enterprise only. List of privileges the user or role has on the user-defined function. For more information, read [User-Defined Function \(UDF\) Privileges](#).

Enum: ALTER, EXECUTE, MANAGE_GRANTS, OWNERSHIP

Example: ["ALTER","EXECUTE"]

Creating a User-Defined Function

Create a new user-defined function.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For user-defined functions, the entityType is `function`.

path Body Array of String

Path where you want to create the user-defined function within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the function itself as the last item in the array.

Example: ["team_folder","test_subfolder","filter_domain_orderdates"]

isScalar Body Boolean

If the user-defined function is a scalar function, `true`. If the user-defined function is a tabular function, `false`.

Example: false

functionArgList Body String

The name of each argument in the user-defined function and the argument's data type. Separate the name and data type with a single space. If the function includes multiple arguments, separate the arguments with a comma.

note

In response objects, the functionArgList value may contain aliases for data types, such as `CHARACTER VARYING` (an alias for `VARCHAR`).

Example: domain VARCHAR, orderdate DATE

functionBody Body String

The statement that the user-defined function should execute.

Example: SELECT name, email, order_date FROM customer_data WHERE LOWER(email) LIKE '%' || LOWER(domain) AND order_date >= orderdate

returnType Body String

The data type of each column that the user-defined function should return.

Example: name VARCHAR, email VARCHAR, order_date DATE

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the user-defined function and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure function-specific access control privileges.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER", "EXECUTE"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER", "EXECUTE"]}]}

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the user-defined function and the specific privileges each user should have.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER", "EXECUTE"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the user-defined function and the specific privileges each role should have.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER", "EXECUTE"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the user-defined function.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the user-defined function. For more information, read [User-Defined Function \(UDF\) Privileges](#).

Enum: ALTER, EXECUTE, MANAGE_GRANTS, OWNERSHIP

Example: ["ALTER","EXECUTE"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "function",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "isScalar": false,
  "functionArgList": "domain VARCHAR, orderdate DATE",
  "functionBody": "SELECT name, email, order_date FROM customer_data WHERE LOWER(email)
LIKE '%' || LOWER(domain) AND order_date >= orderdate",
  "returnType": "name VARCHAR, email VARCHAR, order_date DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWPd7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\"
WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\"
DATE",
  "accessControllList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  },
  "permissions": [],
  "owner": {
    "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
    "ownerType": "USER"
  }
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

Retrieving a User-Defined Function by ID

Retrieve a user-defined function and information about its contents by specifying the function's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user-defined function that you want to retrieve.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/1568aa06-4eac-48cf-bc30-2aa3053c2840' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBwPD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\"
WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\"
DATE",
  "accessControlList": {
```

```

"users": [
  {
    "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
    "permissions": [
      "ALTER",
      "EXECUTE"
    ]
  }
],
"roles": [
  {
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
      "ALTER",
      "EXECUTE"
    ]
  }
]
},
"permissions": [],
"owner": {
  "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a User-Defined Function by Path

Retrieve a user-defined function and information about its contents by specifying the function's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the user-defined function within Dremio. The path consists of the source or

space, followed by any folder and subfolders, followed by the name of the function itself. Separate each level of the path with a forward slash.

Example: team_folder/test_subfolder/filter_domain_orderdates

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/team_folder/test_subfolder/filter_domain_orderdates' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\" WHERE LOWER(\"email\") LIKE '% ' || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\" DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
```

```

        "EXECUTE"
    ]
}
],
"permissions": [],
"owner": {
    "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Updating a User-Defined Function

Update the specified user-defined function.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user-defined function that you want to update.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

entityType Body String

Type of the catalog object. For user-defined functions, the entityType is `function`.

id Body String (UUID)

Unique identifier of the user-defined function that you want to update.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

path Body Array of String

Path of the user-defined function within Dremio, expressed as an array. The path consists of the source or user-defined function, followed by any folder and subfolders,

followed by the name of the function itself as the last item in the array.

Example: ["team_folder","test_subfolder","filter_domain_orderdates"]

tag Body String

Unique identifier of the version of the user-defined function that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the user-defined function.

Example: qBWpD7x6+Ws=

isScalar Body Boolean

If the user-defined function is a scalar function, `true`. If the user-defined function is a tabular function, `false`.

Example: false

functionArgList Body String

The name of each argument in the user-defined function and the argument's data type. Separate the name and data type with a single space. If the function includes multiple arguments, separate the arguments with a comma.

note

In response objects, the functionArgList value may contain aliases for data types, such as `CHARACTER VARYING` (an alias for `VARCHAR`).

Example: domain VARCHAR, orderdate DATE

functionBody Body String

The statement that the user-defined function should execute.

Example: SELECT name, email, phone_number, order_date FROM customer_data WHERE LOWER(email) LIKE '%' || LOWER(domain) AND order_date >= orderdate

returnType Body String

The data type of each column that the user-defined function should return.

Example: name VARCHAR, email VARCHAR, phone_number VARCHAR, order_date DATE

accessControlList Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the user-defined function and the specific privileges each user or role should have. If you omit the accessControlList object in a PUT request, Dremio removes all existing user and role access settings from the function. To keep existing user and role access settings while making other updates, duplicate the existing accessControlList array in the PUT request.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALTER","EXECUTE", "MANAGE_GRANTS"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["ALTER","EXECUTE"]}]}

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the user-defined function and the specific privileges each user should have. If you omit the users object in a PUT request, Dremio removes all existing user access settings from the function. To keep existing user access settings while making other updates, duplicate the existing users array in the PUT request.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER","EXECUTE", "MANAGE_GRANTS"]}]]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the user-defined function and the specific privileges each role should have. If you omit the roles object in a PUT request, Dremio removes all existing role access settings from the function. To keep existing role access settings while making other updates, duplicate the existing roles array in the PUT request.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER","EXECUTE"]}]]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the user-defined function.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the user-defined function. For more information, read [User-Defined Function \(UDF\) Privileges](#).

Enum: ALTER, EXECUTE, MANAGE_GRANTS, OWNERSHIP

Example: ["ALTER","EXECUTE", "MANAGE_GRANTS"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/1568aa06-4eac-48cf-bc30-2aa3053c2840' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
```

```

"isScalar": false,
"functionArgList": "domain VARCHAR, orderdate DATE",
"functionBody": "SELECT name, email, phone_number, order_date FROM customer_data WHERE
LOWER(email) LIKE '%' || LOWER(domain) AND order_date >= orderdate",
"returnType": "name VARCHAR, email VARCHAR, phone_number VARCHAR, order_date DATE",
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER",
        "EXECUTE",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "ALTER",
        "EXECUTE"
      ]
    }
  ]
}
}'

```

Example Response

```

{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "4RuPbmWPoa9=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-07T17:17:17.360Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"phone_number\", \"order_date\" FROM
\"customer_data\" WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >=
orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING,
\"phone_number\" CHARACTER VARYING, \"order_date\" DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [

```

```

        "ALTER",
        "EXECUTE",
        "MANAGE_GRANTS"
    ]
}
],
"roles": [
{
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
        "ALTER",
        "EXECUTE"
    ]
}
]
},
"permissions": [],
"owner": {
    "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
    "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Deleting a User-Defined Function

Delete the specified user-defined function.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user-defined function that you want to delete.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/1568aa06-4eac-48cf-bc30-2aa3053c2840' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Table](/25.x/reference/api/catalog/table)[Next

View](/25.x/reference/api/catalog/view)

User-Defined Function Attributes

Creating a User-Defined Function

Parameters

Response Status Codes

Retrieving a User-Defined Function by ID

Parameters

Response Status Codes

Retrieving a User-Defined Function by Path

Parameters

Response Status Codes

Updating a User-Defined Function

Parameters

Response Status Codes

Deleting a User-Defined Function

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/view>

Version: 25.x

On this page

View

Use the Catalog API to retrieve, create, update, and delete views.

View Object

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
```



```

        "MANAGE_GRANTS"
    ]
}
],
"roles": [
    {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
            "SELECT",
            "ALTER"
        ]
    }
]
},
"permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
],
"owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
},
"fields": [
    {
        "name": "pickup_datetime",
        "type": {
            "name": "TIMESTAMP"
        }
    },
    {
        "name": "passenger_count",
        "type": {
            "name": "BIGINT"
        }
    },
    {
        "name": "passenger_payment_method",
        "type": {

```

```

    "name": "STRUCT",
    "subSchema": [
      {
        "name": "cash",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "credit-debit",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "payment-app",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "other",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
}
]

```

```
}
```

View Attributes

entityType String

Type of the catalog object. For views, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the view.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

type String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

Example: VIRTUAL_DATASET

path Array of String

Path of the view within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the view itself as the last item in the array.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

createdAt String

Date and time that the view was created, in UTC format.

Example: 2022-11-17T18:31:23.236Z

isMetadataExpired Boolean

If true, the metadata of the tables that the view is defined on needs to be refreshed. To refresh it, run the ALTER VIEW command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the view.

If NULL, metadata has never yet been collected for the tables that the view is defined on.

lastMetadataRefreshAt String

Date and time that the metadata of the tables that the view is defined on was last refreshed. In UTC format.

Example: 2024-01-31T09:50:01.012Z

tag String (UUID)

Unique identifier of the version of the view. Dremio changes the tag whenever the view

changes and uses the tag to ensure that PUT requests apply to the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql String

SQL query used to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Array of String

Context for the SQL query used to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Object

Enterprise only. Information about users and roles with access to the view and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if view-specific access control privileges are not set.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT","ALTER"] }, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT","ALTER","MANAGE_GRANTS"] }, {"roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]}]

permissions Array of String

Enterprise-only. List of the privileges that you have on the view. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ","WRITE","ALTER_REFLECTION","SELECT","ALTER","VIEW_REFLECTION","MODIFY","MANAGE_GRANTS","CREATE_TABLE","DROP","EXTERNAL_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE_SOURCE","ALL"]

owner String

Information about the view's owner.

Example: {"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8","ownerType": "USER"}

fields Array of Object

Attributes that represent the dataset schema.

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the view and the specific privileges each

user has.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]]

roles Array of Object

Enterprise only. List of roles whose members have access to the view and the specific privileges each role has.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]]

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the view. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the view's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the view.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the view field.

Example: pickup_datetime

type Object

Information about the view field.

Attributes of the `type` Object

name String

Name of the view field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: TIMESTAMP

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 2

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema is listed only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: cash

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Creating a View

Create a view from a table in Dremio.

Method and URL

POST /api/v3/catalog

Parameters

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the view itself as the last item in the array. The name of the view cannot include the following special characters: `/`, `:`, `[`, or `]`. Views can only be created in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

sql Body String

SQL query to use to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Body Array of String

Context for the SQL query to use to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"]}]}

Parameters of the `accessControlList` Object

users [Body] Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "type": "VIRTUAL_DATASET",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
```



```

        "MANAGE_GRANTS"
    ]
}
],
"roles": [
{
    "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
    "permissions": [
        "SELECT",
        "ALTER"
    ]
}
]
}
}
}'

```

Example Response

```

{
    "entityType": "dataset",
    "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
    "type": "VIRTUAL_DATASET",
    "path": [
        "Business",
        "Transportation",
        "NYC-taxi-trips-short-distance"
    ],
    "createdAt": "2022-11-17T18:31:23.236Z",
    "isMetadataExpired": false,
    "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
    "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
    "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
    "sqlContext": [
        "Samples",
        "samples.dremio.com"
    ],
    "accessControlList": {
        "users": [
            {
                "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
                "permissions": [
                    "SELECT",
                    "ALTER"
                ]
            },
            {
                "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
                "permissions": [
                    "SELECT",
                    "ALTER",
                    "MANAGE_GRANTS"
                ]
            }
        ]
    }
}

```

```

    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "other",

```

```

        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a View by ID

Retrieve a view by specifying the view's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to retrieve.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
```

```

        "ALTER"
    ]
},
{
    "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
    ]
}
],
"roles": [
    {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
            "SELECT",
            "ALTER"
        ]
    }
]
},
"owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
},
"fields": [
    {
        "name": "pickup_datetime",
        "type": {
            "name": "TIMESTAMP"
        }
    },
    {
        "name": "passenger_count",
        "type": {
            "name": "BIGINT"
        }
    },
    {
        "name": "passenger_payment_method",
        "type": {
            "name": "STRUCT",
            "subSchema": [
                {
                    "name": "cash",
                    "type": {
                        "name": "BOOLEAN"
                    }
                }
            ],
            {
                "name": "credit-debit",
                "type": {
                    "name": "BOOLEAN"
                }
            }
        }
    }
]

```

```

    }
  },
  {
    "name": "payment-app",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "other",
    "type": {
      "name": "BOOLEAN"
    }
  }
]
}
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
]
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a View by Path

Retrieve a view by specifying the view's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

View's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" view in the "samples.dremio.com" folder within the space "Transportation," the path is `Transportation/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Business/Transportation/NYC-taxi-trips-short-distance

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Business/Transportation/NYC-taxi-trips-short-distance' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
}
```

```

"createdAt": "2022-11-17T18:31:23.236Z",
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  }
]

```



```

},
{
  "name": "passenger_payment_method",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "cash",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "credit-debit",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "payment-app",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "other",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  }
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {

```

```
    "name": "DOUBLE"
  }
}
]
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Updating a View

Update a view in Dremio.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to update.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the updated view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the name for the view itself as the last item in the array. Views can only be saved in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

tag Body String (UUID) Optional

Unique identifier of the version of the view that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the view. If you do not provide a tag, Dremio automatically updates the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql Body String

SQL query to use to update the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC

sqlContext Body Array of String

Context for the SQL query to use for the updated view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "type": "VIRTUAL_DATASET",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ]
}'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-by-distance"
  ],
  "createdAt": "2023-01-20T15:26:39.780Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "7cab1a42-8835-4d31-827b-fedeelad38d1",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
```

```

        "SELECT",
        "ALTER"
    ]
},
{
    "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
    ]
}
],
"roles": [
    {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
            "SELECT",
            "ALTER"
        ]
    }
]
},
"owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
},
"fields": [
    {
        "name": "pickup_datetime",
        "type": {
            "name": "TIMESTAMP"
        }
    },
    {
        "name": "passenger_count",
        "type": {
            "name": "BIGINT"
        }
    },
    {
        "name": "passenger_payment_method",
        "type": {
            "name": "STRUCT",
            "subSchema": [
                {
                    "name": "cash",
                    "type": {
                        "name": "BOOLEAN"
                    }
                }
            ]
        },
        {
            "name": "credit-debit",
            "type": {

```

```

        "name": "BOOLEAN"
      }
    },
    {
      "name": "payment-app",
      "type": {
        "name": "BOOLEAN"
      }
    },
    {
      "name": "other",
      "type": {
        "name": "BOOLEAN"
      }
    }
  ]
}
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Refreshing the Reflections on a View

Refresh the Reflections associated with the specified view.

Read [Refreshing Reflections](#) to learn how refreshing works.

Method and URL

```
POST /api/v3/catalog/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier for the view you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example Request

```
curl -X POST \
'https://api.dremio.cloud//api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Deleting a View

Delete the specified view.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to delete.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

User-Defined Function](/25.x/reference/api/catalog/user-defined-function)[Next

Lineage](/25.x/reference/api/catalog/lineage)

View Attributes

Creating a View

Parameters

Response Status Codes

Retrieving a View by ID

Parameters

Response Status Codes

Retrieving a View by Path

Parameters

Response Status Codes

Updating a View

Parameters

Response Status Codes

Refreshing the Reflections on a View

Parameters

Response Status Codes

Deleting a View

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/lineage>

Version: 25.x

On this page

Lineage Enterprise

Use the Catalog API to retrieve lineage information about datasets (tables and views). The lineage object includes information about the dataset's sources, parent objects, and child objects.

Lineage Object

```
{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
```

```

    "createdAt": "2022-02-14T21:57:48.794Z"
  },
],
"parents": [
  {
    "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
    "path": [
      "Samples",
      "samples.dremio.com",
      "zips.json"
    ],
    "tag": "MAntohVzwLw=",
    "type": "DATASET",
    "datasetType": "PROMOTED",
    "createdAt": "2023-01-18T18:49:09.669Z"
  },
],
"children": [
  {
    "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
    "path": [
      "@dremio",
      "NYC_zip"
    ],
    "tag": "0WKrfpEKzW4=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-25T02:11:46.344Z"
  },
  {
    "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
    "path": [
      "@dremio",
      "Chicago_zip"
    ],
    "tag": "gsaDW5h4GCs=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-25T00:09:12.461Z"
  }
]
}

```

Lineage Attributes

sources Array of Object

Information about the sources the dataset uses. Each object in the sources array represents one source.

Example: [{"id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2","path": ["Samples"],"tag": "lz1v71CeTQY=","type": "CONTAINER","containerType": "SOURCE","createdAt":

```
"2023-02-14T21:57:48.794Z"}]
```

parents Array of Object

Information about the parent objects for the dataset. Each object in the parents array represents one parent object. If a view represents a join of two other datasets, the parents array includes the two joined datasets. The parents array is empty if the dataset does not have parent objects.

Example:

```
[{"id": "3419fa3a-b5b3-4438-b864-a27ec4e18752", "path": ["Samples", "samples.dremio.com", "zips.json"], "tag": "MAntohVzwLw=", "type": "DATASET", "datasetType": "PROMOTED", "createdAt": "2023-01-18T18:49:09.669Z"}]
```

children Array of Object

Information about other catalog objects that reference the dataset. Each object in the children array represents one child object. The children array is empty if the dataset does not have child objects.

Example:

```
[{"id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75", "path":["@dremio", "NYC\ _zip"], "tag": "OWKrfpEKzW4=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-25T02:11:46.344Z"}, {"id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0", "path":["@dremio", "Chicago\ _zip"], "tag": "gsaDW5h4GCs=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-25T00:09:12.461Z"}]
```

Attributes of Objects in the `sources` Array

id String (UUID)

Unique identifier of the source associated with the dataset.

Example: 21077e5d-fe6f-4a29-843f-58fa3acb17c2

path Array of String

Path of the source within Dremio, expressed as an array.

Example: ["Samples"]

tag String

Unique identifier of the version of the source. Dremio changes the tag whenever the source changes.

Example: lz1v71CeTQY=

type String

Type of source. For sources in lineage responses, the type is `CONTAINER`.

Example: CONTAINER

containerType String

Type of container for the source.

Enum: HOME, SOURCE

Example: SOURCE

createdAt String

Date and time that the source was created, in UTC format. Not included for sources with the containerType `HOME`.

Example: 2022-02-14T21:57:48.794Z

Attributes of Objects in the `parents` Array

id String (UUID)

Unique identifier of the parent object.

Example: 3419fa3a-b5b3-4438-b864-a27ec4e18752

path Array of String

Path of the parent object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the parent object itself as the last item in the array.

Example: ["Samples","samples.dremio.com","zips.json"]

tag String

Unique identifier of the version of the parent object. Dremio changes the tag whenever the parent object changes.

Example: MAntohVzwLw=

type String

Type of parent object. For parent objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the parent object. If the parent object is a table, `PROMOTED`. If the parent object is a view, `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: PROMOTED

createdAt String

Date and time that the parent object was created, in UTC format.

Example: 2023-01-18T18:49:09.669Z

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the child object.

Example: 170e211e-4235-4d8d-acb5-3d4dbfe99c75

path Array of String

Path of the child object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the child object itself as the last item in the array.

Example: ["@dremio","NYC_zip"]

tag String

Unique identifier of the version of the child object. Dremio changes the tag whenever the child object changes.

Example: OWKrfpEKzW4=

type String

Type of child object. For child objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the child object. For child objects in lineage responses, the datasetType is `VIRTUAL`.

Example: VIRTUAL

createdAt String

Date and time that the child object was created, in UTC format.

Example: 2023-01-25T02:11:46.344Z

Retrieving Lineage Information About a Dataset

Retrieve lineage information about the specified dataset.

Method and URL

```
GET /api/v3/catalog/{id}/graph
```

Parameters

id Path String (UUID)

Unique identifier of the dataset whose lineage you want to retrieve.

Example: d69b25a3-31c8-4d55-a7cc-dfee2290779b

Example Request

```
curl -X GET \
'https://{hostname}/api/v3/catalog/d69b25a3-31c8-4d55-a7cc-dfee2290779b/graph' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
      "createdAt": "2022-02-14T21:57:48.794Z"
    }
  ],
  "parents": [
    {
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zips.json"
      ],
      "tag": "MAntohVzwLw=",
      "type": "DATASET",
      "datasetType": "PROMOTED",
      "createdAt": "2023-01-18T18:49:09.669Z"
    }
  ],
  "children": [
    {
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
      "path": [
        "@dremio",
        "NYC_zip"
      ],
      "tag": "OWKrfpEKzW4=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T02:11:46.344Z"
    },
    {
      "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
      "path": [
        "@dremio",
        "Chicago_zip"
      ],
    }
  ]
}
```

```

    "tag": "gsaDW5h4GCs=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-25T00:09:12.461Z"
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

View](/25.x/reference/api/catalog/view)[Next

Tag](/25.x/reference/api/catalog/tag)

Lineage Attributes

Retrieving Lineage Information About a Dataset

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/tag>

Version: 25.x

On this page

Tag

Use the Catalog API to create, update, and retrieve tags.

Tag Object

```

{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ]
}

```

```
],  
  "version": "VJ3ijXH4m6k=",  
}
```

Tag Attributes

tags Array of String

List of tags that apply to the dataset.

Example: ["NYC","taxi","2023"]

version String

Unique identifier of the set of tags. Dremio changes the version whenever any of the tags change and uses the version value to ensure that updates apply to the most recent version of the set of tags.

Example: VJ3ijXH4m6k=

Creating Tags

Create one or more tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset for which you want to add tags.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. Tags are case-insensitive. Each tag can be listed only once for each dataset. Each tag can have a maximum of 128 characters. Tags cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: ["NYC","taxi","2023"]

Example Request

```
curl -X POST  
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json' \  
--data-raw '{
```



```
"tags": ["NYC", "taxi", "2023"]
}'
```

Example Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving Tags

Retrieve the tags applied to the specified dataset.

Method and URL

```
GET /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example Request

```
curl -X GET \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Updating Tags

Update the tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. If you want to keep any of the existing tags, include them in the tags array. Tags are case-insensitive and must be distinct (in other words, list each tag only once for each dataset). Each tag may have a maximum of 128 characters. Tags cannot include the following special characters: ``/`, ``:`, ``[`, or ``\`.

Example: ["NYC","taxi","2023","archived"]

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are updating the most recent version of the tags.

Example: VM3ijXH4m6k=

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023", "archived"],
  "version": "VM3ijXH4m6k="
}'
```

Example Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023",
    "archived"
  ],
  "version": "yiZSE++9wiU="
}
```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Deleting Tags

Delete the tags for the specified dataset.

note

Deleting tags means sending an empty array to replace the existing tags with no tags. The tag object will still exist, but it will contain an empty `tags` array and no tags will appear for the dataset in the Dremio UI.

Method and URL

POST /api/v3/catalog/{dataset-id}/collaboration/tag

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to remove.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

Empty array to represent deletion of all tags for the dataset.

Example: []

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are deleting tags from the most recent version.

Example: yiZSE++9wiU=

Example Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": [],
  "version": "yiZSE++9wiU="
}'
```

Example Response

```
{
  "tags": [],
  "version": "wuTAKuRcVas="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[Previous

Lineage](/25.x/reference/api/catalog/lineage)[Next

Wiki](/25.x/reference/api/catalog/wiki)

Tag Attributes

Creating Tags

Parameters

Response Status Codes

Retrieving Tags

Parameters

Response Status Codes

Updating Tags

Parameters

Response Status Codes

Deleting Tags

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/wiki>

Version: 25.x

On this page

Wiki

Use the Catalog API to create, update, and retrieve the wiki for a source, space, or dataset.

Wiki Object

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
```

```

Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 4
}

```

Wiki Attributes

text String

Text displayed in the wiki, formatted with [GitHub-flavored Markdown](#).

version Integer

Number for the most recent version of the wiki, starting with `0`. Dremio increments the value by 1 each time the wiki changes and uses the version value to ensure that updates apply to the most recent version of the wiki.

Example: 4

Creating a Wiki

Create a wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset for which you want to add the wiki.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki. Use [GitHub-flavored Markdown](#) for wiki formatting and `\\n` for new lines and blank lines. Each wiki may have a maximum of 100,000 characters.

Example Request

```
curl -X POST
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum."
}'
```

Example Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a Wiki

Retrieve the wiki for the specified source, space, or dataset.

Method and URL

```
GET /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example Request

```
curl -X GET \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio. Here is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n### Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Updating a Wiki

Update the wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki, formatted with [GitHub-flavored Markdown](#).

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are updating the most recent version of the wiki.

Example: 0

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is an update to the bulleted list:\n* An item in
a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted
list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered
list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere
is a sentence that includes an [external link to
https://dremio.com](https://dremio.com).\n\nHere is an
```

```
image:\n\n\n\nUt  
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea  
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum  
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in  
culpa qui officia deserunt mollit anim id est laborum.",  
  "version": 0  
'}
```

Example Response

```
{  
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio.  
Here is some text in bold. Here is some text in italics.\n\nHere is an example  
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur  
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna  
aliqua.\n\n\n## Heading Level 2\n\nHere is an update to the bulleted list:\n* An item in  
a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted  
list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered  
list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere  
is a sentence that includes an [external link to  
https://dremio.com](https://dremio.com).\n\nHere is an  
image:\n\n\n\nUt  
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea  
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum  
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in  
culpa qui officia deserunt mollit anim id est laborum.",  
  "version": 1  
}
```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Deleting a Wiki

Delete the wiki for the specified source, space, or dataset.

note

Deleting the wiki entails sending an empty string to replace the existing wiki with no wiki. The wiki object will still exist, but it will contain an empty `text` value and no wiki

will appear for the source, space, or dataset in the Dremio UI.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to delete.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Empty string to represent deletion of the wiki.

Example: ""

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are deleting the most recent version of the wiki.

Example: 1

Example Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "",
  "version": 1
}'
```

Example Response

```
{
  "text": "",
  "version": 2
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[Previous

Tag]/(25.x/reference/api/catalog/tag)[Next

Privileges]/(25.x/reference/api/catalog/privileges)

Wiki Attributes

Creating a Wiki

Parameters

Response Status Codes

Retrieving a Wiki

Parameters

Response Status Codes

Updating a Wiki

Parameters

Response Status Codes

Deleting a Wiki

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/privileges>

Version: 25.x

On this page

Privileges Enterprise

caution

The Catalog API Privileges endpoint is deprecated. We expect to remove it by July 2025. In place of the Privileges endpoint, use the Catalog API Grants endpoint to retrieve

privileges and grantees on specific catalog objects.

Use the Catalog API to retrieve information about available privileges on the different types of catalog objects.

Privileges Object

```
{
  "availablePrivileges": [
    {
      "grantType": "SPACE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
      ]
    },
    {
      "grantType": "SOURCE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
      ]
    },
    {
      "grantType": "MUTABLE_SOURCE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "CREATE_TABLE",
        "DELETE",
        "DROP",
        "INSERT",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "TRUNCATE",
        "UPDATE",
        "VIEW_REFLECTION"
      ]
    },
    {
      "grantType": "ARP_SOURCE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "EXTERNAL_QUERY",
        "MANAGE_GRANTS",
```

```

        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER_IN_MUTABLE_SOURCE",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "CREATE_TABLE",
        "DELETE",
        "DROP",
        "INSERT",
        "MANAGE_GRANTS",
        "SELECT",
        "TRUNCATE",
        "UPDATE",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "PDS",
    "privileges": [
        "ALTER",
        "DELETE",
        "INSERT",
        "MANAGE_GRANTS",
        "SELECT",
        "TRUNCATE",
        "UPDATE"
    ]
},
{
    "grantType": "VDS",
    "privileges": [
        "ALTER",
        "MANAGE_GRANTS",
        "SELECT"
    ]
},
{
    "grantType": "FUNCTION",
    "privileges": [

```

```

        "ALTER",
        "EXECUTE",
        "MANAGE_GRANTS",
        "MODIFY"
    ]
}
]
}

```

Privileges Attributes

availablePrivileges Array of Object

Information about the grant types and privileges that are available to assign to users and roles for each type of object in the catalog. Each availablePrivileges object contains two attributes: grantType and privileges.

Example:

```

[{"grantType": "SPACE", "privileges":
["ALTER", "ALTER\_REFLECTION", "MANAGE\ GRANTS", "MODIFY", "SELECT", "VIEW\ REFLE
CTION"]}, {"grantType": "SOURCE", "privileges":
["ALTER", "ALTER\_REFLECTION", "MANAGE\ GRANTS", "MODIFY", "SELECT", "VIEW\ REFLE
CTION"]}, {"grantType": "MUTABLE\ SOURCE", "privileges":
["ALTER", "ALTER\_REFLECTION", "CREATE\ TABLE", "DELETE", "DROP", "INSERT", "MANAGE
\ GRANTS", "MODIFY", "SELECT", "TRUNCATE", "UPDATE", "VIEW\ REFLECTION"]}, {"grantT
ype": "ARP\ SOURCE", "privileges":
["ALTER", "ALTER\_REFLECTION", "EXTERNAL\ QUERY", "MANAGE\ GRANTS", "MODIFY", "S
ELECT", "VIEW\ REFLECTION"]}, {"grantType":
"FOLDER\ IN\ MUTABLE\ SOURCE", "privileges":
["ALTER", "ALTER\_REFLECTION", "CREATE\ TABLE", "DELETE", "DROP", "INSERT", "MANAGE
\ GRANTS", "SELECT", "TRUNCATE", "UPDATE", "VIEW\ REFLECTION"]}, {"grantType":
"FOLDER", "privileges":
["ALTER", "ALTER\_REFLECTION", "MANAGE\ GRANTS", "SELECT", "VIEW\ REFLECTION"]},
{"grantType": "PDS", "privileges":
["ALTER", "DELETE", "INSERT", "MANAGE\ GRANTS", "SELECT", "TRUNCATE", "UPDATE"]}, {"
grantType": "VDS", "privileges":
["ALTER", "MANAGE\ GRANTS", "SELECT"]}, {"grantType": "FUNCTION", "privileges":
["ALTER", "EXECUTE", "MANAGE\ GRANTS", "MODIFY"]}

```

Attributes of Objects in the `availablePrivileges` Array

grantType String

Type of the catalog object on which the listed privileges are available. `ARP_SOURCE` refers to relational-database sources.

Enum: SPACE, SOURCE, MUTABLE\ SOURCE, ARP\ SOURCE, FOLDER\ IN\ MUTABLE\ SOURCE, FOLDER, PDS, VDS, FUNCTION

Example: SPACE

privileges Array of String

List of available privileges on the type of the catalog object specified in grantType. For more information, read [Privileges](#).

Example:

```
["ALTER","ALTER\_REFLECTION","MANAGE\_GRANTS","MODIFY","SELECT","VIEW\_REFLECTION"]
```

Retrieving All Catalog Privileges

Retrieve information about the available privileges on each type of object in the catalog.

caution

This endpoint is deprecated. We expect to remove it by July 2025.

In place of this endpoint, use the Catalog API [Grants](#) endpoint to retrieve privileges and grantees on specific catalog objects.

Method and URL

```
GET /api/v3/catalog/privileges
```

Parameters

type Query String Optional

Type of the catalog object whose available privileges you want to retrieve. For more information, read [type Query Parameter](#).

Enum: SPACE, SOURCE, MUTABLE_SOURCE, ARP_SOURCE, FOLDER_IN_MUTABLE_SOURCE, FOLDER, PDS, VDS, FUNCTION

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/privileges' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "availablePrivileges": [
    {
      "grantType": "SPACE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
      ]
    }
  ]
}
```



```

},
{
  "grantType": "SOURCE",
  "privileges": [
    "ALTER",
    "ALTER_REFLECTION",
    "MANAGE_GRANTS",
    "MODIFY",
    "SELECT",
    "VIEW_REFLECTION"
  ]
},
{
  "grantType": "MUTABLE_SOURCE",
  "privileges": [
    "ALTER",
    "ALTER_REFLECTION",
    "CREATE_TABLE",
    "DELETE",
    "DROP",
    "INSERT",
    "MANAGE_GRANTS",
    "MODIFY",
    "SELECT",
    "TRUNCATE",
    "UPDATE",
    "VIEW_REFLECTION"
  ]
},
{
  "grantType": "ARP_SOURCE",
  "privileges": [
    "ALTER",
    "ALTER_REFLECTION",
    "EXTERNAL_QUERY",
    "MANAGE_GRANTS",
    "MODIFY",
    "SELECT",
    "VIEW_REFLECTION"
  ]
},
{
  "grantType": "FOLDER_IN_MUTABLE_SOURCE",
  "privileges": [
    "ALTER",
    "ALTER_REFLECTION",
    "CREATE_TABLE",
    "DELETE",
    "DROP",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE",

```

```

        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "PDS",
    "privileges": [
        "ALTER",
        "DELETE",
        "INSERT",
        "MANAGE_GRANTS",
        "SELECT",
        "TRUNCATE",
        "UPDATE"
    ]
},
{
    "grantType": "VDS",
    "privileges": [
        "ALTER",
        "MANAGE_GRANTS",
        "SELECT"
    ]
},
{
    "grantType": "FUNCTION",
    "privileges": [
        "ALTER",
        "EXECUTE",
        "MANAGE_GRANTS",
        "MODIFY"
    ]
}
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Wiki](/25.x/reference/api/catalog/wiki)[Next

Grants](/25.x/reference/api/catalog/grants)

Privileges Attributes

Retrieving All Catalog Privileges

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/grants>

Version: 25.x

On this page

Grants Enterprise

Use the Catalog API to grant user and role privileges on specific catalog objects.

Grants Object

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
      "firstName": "Jean",
      "lastName": "Smith",
    }
  ]
}
```

```

    "email": "jean_smith@example.com"
  },
  {
    "privileges": [
      "ALTER",
      "SELECT"
    ],
    "granteeType": "ROLE",
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "name": "examplerole"
  }
]
}

```

Grants Attributes

id String

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

availablePrivileges Array of String

List of available privileges on the catalog object.

Example:

["ALTER","DELETE","INSERT","MANAGE_GRANTS","SELECT","TRUNCATE","UPDATE"]

grants Array of Object

Information about the privileges and grantees for the catalog object. If the grants array is empty, there are no explicit grants for the object.

note

An empty grants array does not mean no users have access to the object at all. For example, admin users implicitly have all privileges on all catalog objects, owners implicitly have all privileges on everything they own, and children objects inherit the grants for their parent objects.

Example: [{"privileges": ["ALTER","SELECT","MANAGE_GRANTS"],"granteeType": "USER","id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0","name": "jeansmith","firstName": "Jean","lastName": "Smith","email": "jean_smith@example.com"}, {"privileges": ["ALTER","SELECT"],"granteeType": "ROLE","id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","name": "examplerole"}]

Attributes of Objects in the `grants` Array

privileges String

List of privileges granted to the user or role. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE_GRANTS"]

granteeType String

Type of grantee.

Enum: USER, ROLE

Example: USER

id String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

name String

Name of the user or role.

Example: jeansmith

firstName String

For users, the user's first name. Not included for roles.

Example: Jean

lastName String

For users, the user's last name. Not included for roles.

Example: Smith

email String

For users, the user's email address. Not included for roles.

Example: jean_smith@example.com

Creating or Updating Privilege Grants on a Catalog Object

Create or update the privileges granted to users and roles on the specified catalog object.

note

You must have the MANAGE GRANTS privilege to create or update privilege grants on catalog objects.

Method and URL

```
PUT /api/v3/catalog/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

grants Body Array of Object

Array of objects that specify which users and roles should have privileges on the catalog object, as well as each user's and role's specific privileges. May include objects for users, roles, or both.

Example: [{"privileges": ["ALTER","SELECT","MANAGE_GRANTS"],"granteeType": "USER","id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"}, {"privileges": ["SELECT","ALTER"],"granteeType": "ROLE","id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"}]

Parameters of Objects in the `grants` Array

privileges Body Array of String

List of privileges to grant to the user or role. Use the [Privileges](#) endpoint to retrieve a list of available privileges on the catalog object type. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE_GRANTS"]

granteeType Body String

Type of grantee.

Enum: USER, ROLE

Example: USER

id Body String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE\_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"
    }
  ]
}
```

```
    },
    {
      "privileges": [
        "SELECT",
        "ALTER"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"
    }
  ]
}'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving Privileges and Grantees on a Catalog Object

Retrieve information about the privileges granted to users and roles on the specified catalog object.

note

Use this endpoint in place of the Catalog API [Privileges](#) endpoint, which is deprecated. We expect to remove the Privileges endpoint by July 2025.

You must have the [MANAGE GRANTS privilege](#) to retrieve privilege grants on catalog objects.

Method and URL

GET /api/v3/catalog/{id}/grants

Parameters

id Path String (UUID)

Unique identifier of the object whose privilege grants you want to retrieve.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
      "firstName": "Jean",
      "lastName": "Smith",
      "email": "jean_smith@example.com"
    },
    {
      "privileges": [
        "ALTER",
        "SELECT"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "name": "examplerole"
    }
  ]
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Privileges](/25.x/reference/api/catalog/privileges)[Next

Dataset](/25.x/reference/api/datasets/)

Grants Attributes

Creating or Updating Privilege Grants on a Catalog Object

Parameters

Response Status Codes

Retrieving Privileges and Grantees on a Catalog Object

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/container-space/>

Version: 25.x

On this page

Space

Use the Catalog API to retrieve information about spaces and the child objects they contain, as well as to create, update, and delete spaces.

Space Object

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
```

```

        "testfolder"
    ],
    "tag": "0McuCL4MzBU=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
    "path": [
        "Example-Space",
        "travel_testing"
    ],
    "tag": "i4mn1SmHqVM=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
},
{
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
        "Example-Space",
        "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "MODIFY"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "MODIFY"
            ]
        }
    ]
},
"permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",

```

```

    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Space Attributes

entityType String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id String (UUID)

Unique identifier of the space.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name String

Name of the space.

Example: Example-Space

tag String

Unique identifier of the version of the space. Dremio changes the tag whenever the space changes and uses the tag to ensure that PUT requests apply to the most recent version of the space.

Example: zzOQfjY9IU0=

createdAt String

Date and time that the space was created, in UTC format.

Example: 2023-01-12T18:44:43.237Z

children Array of Object

Information about each catalog object in the space.

```
Example: [{"id": "8da037a1-8e50-422b-9a2b-cafb03f57c71", "path":
["Example-Space", "testfolder"], "tag": "0McuCL4MzBU=", "type":
"CONTAINER", "containerType": "FOLDER"}, {"id":
"f32dfe85-32e2-4c31-b2b4-bfd62ab3f473", "path":
["Example-Space", "travel\\_testing"], "tag": "i4mnlSmHqVM=", "type":
"DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-30T17:54:25.547Z"}, {"id":
"7f1c4660-cd7b-40d0-97d1-b8a6f431cbda", "path": ["Example-Space", "zips"], "tag":
"ITlp8+qyIMQ=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt":
"2023-02-08T16:24:25.084Z"}]
```

accessControlList Object

Enterprise only. Information about users and roles with access to the space and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if space-specific access control privileges are not set.

```
Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions":
["MODIFY"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions":
["MODIFY"]}]}
```

permissions Array of String

Enterprise-only. List of the privileges that you have on the space. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

```
Example:
["READ", "WRITE", "ALTER\\_REFLECTION", "SELECT", "ALTER", "VIEW\\_REFLECTION", "MODI
FY", "MANAGE\\_GRANTS", "CREATE\\_TABLE", "DROP", "EXTERNAL\\_QUERY", "INSERT", "TRU
NCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE\\_SOURCE", "ALL"]
```

owner Object

Information about the space's owner.

```
Example: {"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43", "ownerType":
"USER"}
```

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the catalog object.

Example: 8da037a1-8e50-422b-9a2b-cafb03f57c71

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["Example-Space", "testfolder"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the object.

Example: 0McuCL4MzBU=

type String

Type of the catalog object.

Enum: CONTAINER, DATASET, FILE

Example: CONTAINER

containerType String

For catalog entities with the type `CONTAINER`, the type of container.

Enum: FOLDER, FUNCTION

Example: FOLDER

datasetType String

For catalog objects in a space with the type `DATASET`, the datasetType is `VIRTUAL` (spaces cannot contain tables, only views).

Example: VIRTUAL

createdAt String

For catalog objects in a space with the type `DATASET`, date and time that the catalog object was created, in UTC format.

Example: 2023-01-30T17:54:25.547Z

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the space and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the space and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the space's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the space.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise only. Unique identifier of the user or role with access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise only. List of privileges the user or role has on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["MODIFY"]

Creating a Space

Create a new space.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

name Body String

Name of the space. The name cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: Example-Space

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure space-specific access control privileges.

Example: `{"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] }], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] }] }`

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have.

Example: `[{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] }]`

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have.

Example: `[{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] }]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: `["MODIFY"]`

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
```

```
--data-row '{
  "entityType": "space",
  "name": "Example-Space",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "permissions": [],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}
```



```
}  
}
```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Retrieving a Space by ID

Retrieve a space and information about its contents by specifying the space's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to retrieve.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in

concert with the `pageToken` query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: `?maxChildren=25`

`pageToken` Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the `maxChildren` query parameter: the first request URL includes `maxChildren` set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified `maxChildren` value, the response includes a `nextPageToken` attribute. Add the `pageToken` query parameter with the `nextPageToken` value to the request URL to retrieve the next page of results. Do not remove or change the `maxChildren` query parameter when you add `pageToken` to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

`?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE`

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McUCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    }
  ],
}
```

```

{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "path": [
    "Example-Space",
    "zips"
  ],
  "tag": "ITlp8+qyIMQ=",
  "type": "DATASET",
  "datasetType": "VIRTUAL",
  "createdAt": "2023-02-08T16:24:25.084Z"
},
{
  "accessControllist": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Response Status Codes

- 200 OK
- 204 No Content
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Retrieving a Space by Path

Retrieve a space and information about its contents by specifying the space's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the space that you want to retrieve. The path is the name of the space.

Example: Example-Space

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Example-Space' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
        "zips"
      ],
      "tag": "ITlp8+qyIMQ=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-08T16:24:25.084Z"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
```

```

        "MODIFY"
    ]
}
],
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Updating a Space

Update the specified space.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

entityType Body String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id Body String (UUID)

Unique identifier of the space to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name Body String

Name of the space to update.

Example: Example-Space

tag Body String

Unique identifier of the version of the space that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the space.

Example: zzOQfjY9IU0=

accessControlList Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. If you omit the accessControlList object in a PUT request, Dremio removes all existing user and role access settings from the space. To keep existing user and role access settings while making other updates, duplicate the existing accessControlList array in the PUT request.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALL"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["MODIFY"]}]}

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have. If you omit the users object in a PUT request, Dremio removes all existing user access settings from the space. To keep existing user access settings while making other updates, duplicate the existing users array in the PUT request.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have. If you omit the roles object in a PUT request, Dremio removes all existing role access settings from the space. To keep existing role access settings while making other updates, duplicate the existing roles array in the PUT request.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["ALL"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALL"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "PwZ6e/axHUY=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ]
    }
  ]
}
```



```

    ],
    "tag": "0McucL4MzBU=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
    "path": [
      "Example-Space",
      "travel_testing"
    ],
    "tag": "i4mnlSmHqVM=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
  },
  {
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
      "Example-Space",
      "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER_REFLECTION",
        "MODIFY",
        "ALTER",
        "MANAGE_GRANTS",
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}

```

```
}  
}
```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Deleting a Space

Delete the specified space, including all of the space's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to delete.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0'  
\br/>--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Home](/25.x/reference/api/catalog/container-home)[Next

Folder](/25.x/reference/api/catalog/container-folder)

Space Attributes

Creating a Space

Parameters

Response Status Codes

Retrieving a Space by ID

Parameters

Response Status Codes

Retrieving a Space by Path

Parameters

Response Status Codes

Updating a Space

Parameters

Response Status Codes

Deleting a Space

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/container-folder/>

Version: 25.x

On this page

Folder

Use the Catalog API to retrieve information about folders and the child objects they

contain, as well as to create, update, and delete folders.

Folder Object

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
      ],
      "type": "FILE"
    },
    {
      "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "type": "DATASET",
      "datasetType": "PROMOTED"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "roles": [
    {
```

```

    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
      "SELECT"
    ]
  },
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Folder Attributes

entityType String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

id String

Unique identifier of the folder. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

path Array of String

Path of the folder within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target folder itself as

the last item in the array.

Example: ["Samples","samples.dremio.com"]

tag String

Unique identifier of the version of the folder. Dremio changes the tag whenever the folder changes and uses the tag to ensure that PUT requests apply to the most recent version of the folder.

Example: pRmJ0BQ9SFw=

children Array of Object

Information about each catalog object in the folder.

Example: [{"id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv","path": ["Samples","samples.dremio.com","zip_lookup.csv"],"type": "FILE"}, {"id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg","path": ["Samples","samples.dremio.com","NYC-taxi-trips-iceberg"],"type": "CONTAINER","containerType": "FOLDER"}, {"id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d","path": ["Samples","samples.dremio.com","NYC-taxi-trips"],"type": "DATASET","datasetType": "PROMOTED"}]

accessControlList Object

Enterprise-only. Information about users and roles with access to the folder and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if folder-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the folder. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ","WRITE","ALTER_REFLECTION","SELECT","ALTER","VIEW_REFLECTION","MODIFY","MANAGE_GRANTS","CREATE_TABLE","DROP","EXTERNAL_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE_SOURCE","ALL"]

owner Object

Information about the folder's owner.

Example: {"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43","ownerType": "USER"}

Attributes of Objects in the `children` Array

id String

Unique identifier of the catalog object. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: `dremio:/Samples/samples.dremio.com/zip_lookup.csv`

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: `["Samples","samples.dremio.com","zip_lookup.csv"]`

type String

Type of the catalog object. If the object is saved within a space (including the home space), valid types are ``CONTAINER`` and ``DATASET``. If the object is saved within a source, valid types are ``CONTAINER``, ``FILE``, and ``DATASET``.

Example: `CONTAINER`

containerType String

For catalog objects with the type ``CONTAINER``, the containerType is ``FOLDER``.

Example: `FOLDER`

datasetType String

For catalog objects with the type ``DATASET``, the type of dataset. For tables, the datasetType is ``PROMOTED``. For views, the datasetType is ``VIRTUAL``.

Enum: `PROMOTED`, `VIRTUAL`

Example: `VIRTUAL`

createdAt String

Date and time that the catalog object was created, in UTC format. The createdAt attribute is included only for ``DATASET`` catalog objects that are saved in folders within spaces, not within sources.

Example: `2023-01-30T17:54:25.547Z`

Attributes of the ``accessControlList`` Object

users String

Enterprise-only. List of users with access to the folder and the specific privileges each user has.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}]`

roles String

Enterprise-only. List of roles whose members have access to the folder and the specific privileges each role has.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]`

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the folder's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the folder.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise-only. Unique identifier of the user or role with access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise-only. List of privileges the user or role has on the folder. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Creating a Folder

Create a new folder within a space.

note

The Catalog API cannot create new folders within sources.

Method and URL

POST /api/v3/catalog

Parameters

entityType Body String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

path Body Array of String

Path of the location where the folder should be created within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the target folder itself as the last item in the array. The name of the folder cannot include the following special characters: `/`, `:`, `[`, or ` `.

Example: ["Example-Space","First-Folder","New-Folder"]

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure folder-specific access control privileges.

Example: {"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["SELECT","ALTER"]}], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]}

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["SELECT","ALTER"]}]

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more

information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["SELECT","ALTER"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {

```

```

    "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
    "permissions": [
      "SELECT",
      "ALTER"
    ]
  },
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Retrieving a Folder by ID

Retrieve a folder and information about its contents by specifying the folder's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to retrieve. If the ID is a text path, use URL

encoding to replace any special characters with their UTF-8-equivalent characters, such as `%3A` for a colon; `%2F` for a forward slash; and `%20` for a space. For example, if the ID value is `dremio:/Samples/samples.dremio.com/Dremio University`, the URI-encoded ID is `dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for folders in filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/d4c2a8ba-a972-4db4-8deb-67e1ade684d1' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
      ],
      "type": "FILE"
    },
    {
      "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
      "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "type": "DATASET",
      "datasetType": "PROMOTED"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
```

```

        "SELECT"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a Folder by Path

Retrieve a folder and information about its contents by specifying the folder's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the folder that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Samples/samples.dremio.com

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdkyBDUNoJvwrEXAMPLE

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
```

```

        "samples.dremio.com",
        "zip_lookup.csv"
    ],
    "type": "FILE"
},
{
    "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
    "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
    "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
    ],
    "type": "DATASET",
    "datasetType": "PROMOTED"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "SELECT",
                "ALTER"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "SELECT"
            ]
        }
    ]
},
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found

Updating a Folder

Update the specified folder.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

entityType Body String

Type of the catalog object. For folders, the entityType is ``folder``.

Example: folder

id Body String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

path Body Array of String

Path of the location where the folder is saved within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the target folder itself as the last item in the array.

Example: ["Example-Space", "First-Folder", "New-Folder"]

tag Body String

Unique identifier of the version of the folder that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the folder.

Example: R7COubQq8KE=

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. If you omit the `accessControlList` object in a PUT request, Dremio removes all existing user and role access settings from the folder. To keep existing user and role access settings while making other updates, duplicate the existing `accessControlList` array in the PUT request.

Example: `{"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]}`

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have. If you omit the `users` object in a PUT request, Dremio removes all existing user access settings from the folder. To keep existing user access settings while making other updates, duplicate the existing `users` array in the PUT request.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"]}]`

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have. If you omit the `roles` object in a PUT request, Dremio removes all existing role access settings from the folder. To keep existing role access settings while making other updates, duplicate the existing `roles` array in the PUT request.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["ALL"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALL"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "vnRnYLLpCFU=",
  "children": [
    {
      "id": "d60f9258-e55a-4fc3-97b3-58c6720a70fc",
      "path": [
        "Example-Space",
        "First-Folder",
        "New-Folder",

```

```

        "NYC-trips-weather"
    ],
    "tag": "IHXU70xs80c=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "acba8595-bfcf-4126-887c-d2a19b5afb1d",
    "path": [
        "Example-Space",
        "First-Folder",
        "New-Folder",
        "short-distance-trips"
    ],
    "tag": "KYs/Qywlok8=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-09T19:09:58.789Z"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "ALTER_REFLECTION",
                "ALTER",
                "MANAGE_GRANTS",
                "VIEW_REFLECTION",
                "SELECT"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "SELECT"
            ]
        }
    ]
},
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Deleting a Folder

Delete the specified folder, including all of the folder's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to delete. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: 598697c2-8be0-4050-9731-53563977a17d

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/598697c2-8be0-4050-9731-53563977a17d' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content
401 Unauthorized
403 Forbidden
404 Not Found

Was this page helpful?

[Previous

Space](/25.x/reference/api/catalog/container-space)[Next

File](/25.x/reference/api/catalog/file)

Folder Attributes

Creating a Folder

Parameters

Response Status Codes

Retrieving a Folder by ID

Parameters

Response Status Codes

Retrieving a Folder by Path

Parameters

Response Status Codes

Updating a Folder

Parameters

Response Status Codes

Deleting a Folder

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/file/>

Version: 25.x

On this page

File

Use the Catalog API to retrieve information about formatting data to a table.

File Object

```
{
  "entityType": "file",
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
  "path": [
    "Samples",
```

```
"samples.dremio.com",  
"SF weather 2018-2019.csv"  
]  
}
```

File Attributes

entityType String

Type of the catalog object. For files, the entityType is `file`.

Example: file

id String

Unique identifier of the file. For files, the ID is the text path of the file within Dremio.

Example: dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv

path Array of String

Path of the file within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target file itself as the last item in the array.

Example: ["Samples","samples.dremio.com","SF weather 2018-2019.csv"]

Retrieving a File by Path

Retrieve information about a file by specifying its path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the file that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, replace colons with `%3A` and replace spaces with `%20`.

Example: Samples/samples.dremio.com/SF%20weather%202018-2019.csv

Example Request

```
curl -X GET  
'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/SF%20weather%2018-  
2019.csv' \
```

```
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{  
  "entityType": "file",  
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",  
  "path": [  
    "Samples",  
    "samples.dremio.com",  
    "SF weather 2018-2019.csv"  
  ]  
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Was this page helpful?

[Previous

Folder](/25.x/reference/api/catalog/container-folder)[Next

Table](/25.x/reference/api/catalog/table)

File Attributes

Retrieving a File by Path

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/table/>

Version: 25.x

On this page

Table

Use the Catalog API to retrieve [tables](#), format files and folders as tables, update and refresh tables, and revert tables to files and folders.

Table Object

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "SCHEDULE",
    "refreshPeriodMs": 3600000,
    "gracePeriodMs": 10800000,
    "refreshSchedule": "0 0 8 * * ?",
    "method": "FULL",
    "neverExpire": false,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}
```

```

    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",
  "TRUNCATE",
  "DELETE",
  "UPDATE",
  "EXECUTE",
  "CREATE_SOURCE",
  "ALL"
],
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {

```

```

    "name": "VARCHAR"
  }
},
{
  "name": "city",
  "type": {
    "name": "VARCHAR"
  }
},
{
  "name": "state",
  "type": {
    "name": "VARCHAR"
  }
},
{
  "name": "categories",
  "type": {
    "name": "LIST",
    "subSchema": [
      {
        "type": {
          "name": "VARCHAR"
        }
      }
    ]
  }
},
{
  "name": "review_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "stars",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ]
        }
      }
    ]
  }
}

```

```

        }
    },
    {
        "name": "street",
        "type": {
            "name": "BOOLEAN"
        }
    },
    {
        "name": "lot",
        "type": {
            "name": "BOOLEAN"
        }
    },
    {
        "name": "valet",
        "type": {
            "name": "BOOLEAN"
        }
    }
]
}
},
{
    "name": "Accepts Credit Cards",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Wheelchair Accessible",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Price Range",
    "type": {
        "name": "BIGINT"
    }
}
]
}
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],
"approximateStatisticsAllowed": false
}

```

Table Attributes

entityType String

Type of the catalog object. For tables, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the table.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

type String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example: PHYSICAL_DATASET

path Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples","samples.dremio.com","Dremio University","restaurant_reviews.parquet"]

createdAt String

Date and time that the table was created, in UTC format.

Example: 2024-01-13T19:52:01.894Z

tag String (UUID)

Unique identifier of the version of the table. Dremio changes the tag whenever the table changes and uses the tag to ensure that PUT requests apply to the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

accelerationRefreshPolicy String

Attributes that define the acceleration refresh policy for the table.

isMetadataExpired Boolean

If true, the metadata of the table needs to be refreshed. To refresh it, run the ALTER TABLE command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the table.

If NULL, metadata has never yet been collected for the table.

lastMetadataRefreshAt String

Date and time that the table metadata was last refreshed. In UTC format. If NULL, the

metadata has never yet been refreshed.

Example: 2024-01-31T09:50:01.012Z

format Object

Table format attributes.

accessControlList Object

Enterprise only. Information about users and roles with access to the table and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if table-specific access control privileges are not set.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the table. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ", "WRITE", "ALTER\ REFLECTION", "SELECT", "ALTER", "VIEW\ REFLECTION", "MODIFY", "MANAGE\ GRANTS", "CREATE\ TABLE", "DROP", "EXTERNAL\ QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE\ SOURCE", "ALL"]

owner String

Information about the table's owner.

fields Object

Attributes that represent the table schema.

approximateStatisticsAllowed Boolean

If true, `COUNT DISTINCT` queries run on the table return approximate results. Otherwise, false.

Example: {"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8", "ownerType": "USER"}

Attributes of the `accelerationRefreshPolicy` Object

activePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `neverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

``PERIOD``: Default. The Reflections are refreshed at the end of every period that is defined by `refreshPeriodMs`.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency.

`refreshPeriodMs` Integer

Refresh period for the data in all Reflections for the table, in milliseconds.

Example: 3600000

`refreshSchedule` String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
?	Equivalent to *.
,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

``0 0 0 * * ?`` : Refreshes every day at midnight.

``0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

``0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

`gracePeriodMs` Integer

Maximum age allowed for Reflection data used to accelerate queries, in milliseconds.

Example: 10800000

method String

Approach used for refreshing the data in Reflections defined on tables that are not in the Apache Iceberg format. For more information, read [Refreshing Reflections](#).

Enum: FULL, INCREMENTAL

Example: FULL

refreshField String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Boolean

If the Reflection never expires, the value is `true`. Otherwise, the value is `false`.

Example: false

neverRefresh Boolean

If the Reflection never refreshes, the value is `true`. Otherwise, the value is `false`.

Example: false

sourceRefreshOnDataChanges Boolean

If the table's source is configured so that Reflections on tables in Iceberg format in the source will refresh when new snapshots are created after an update, `true`. Otherwise, `false`.

Attributes of the `format` Object

type String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

Example: Parquet

name String

Table name. Dremio automatically duplicates the name of the origin file or folder to populate this value. The name of the origin file or folder cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: restaurant_reviews.parquet

fullPath Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

ctime Integer

Not used. Has the value `0`.

Example: 0

isFolder Boolean

If the value is `true`, the table was created from a folder. If the value is `false`, the table was created from a file.

Example: false

location String

Location, expressed as a string, where the table's metadata is stored within a Dremio source or space.

Example: /samples.dremio.com/Dremio University/restaurant_reviews.parquet

ignoreOtherFileFormats Boolean

If true, Dremio ignores all non-Parquet files in the related folder structure, and the promoted table works as if only Parquet files are in the folder structure. Otherwise, false. Included only for Parquet folders.

Example: false

metaStoreType String

Not used. Has the value `HDFS`.

Example: HDFS

parquetDataFormat Object

Information about data format for Parquet tables.

dataFormatTypeList Array of String

List of data format types in the table. Included only for Iceberg tables, and `PARQUET` is the only valid value.

Example: ["PARQUET"]

sheetName String

For tables created from files that contain multiple sheets, the name of the sheet used to create the table.

Example: location_1

extractHeader Boolean

For tables created from files, the value is `true` if Dremio extracted the table's column names from the first line of the file. Otherwise, the value is `false`.

Example: false

hasMergedCells Boolean

For tables created from files, the value is `true` if Dremio expanded merged cells in the file when creating the table. Otherwise, the value is `false`.

Example: true

fieldDelimiter String

Character used to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character.

quote String

Character used for quotation marks in the table. May be `"` for a double quote (default), `'` for a single quote, or a custom character.

comment String

Character used to indicate comments in the table. May be `#` for a number sign (default) or a custom character.

escape String

Character used to indicate an escape in the table. May be `"` for a double quote (default), ``` for a back quote, `\` for a backward slash, or a custom character.

lineDelimiter String

Character used to indicate separate lines in the table. May be `\r\n` for a carriage return and a new line (default), `\n` for a new line, or a custom character.

skipFirstLine Boolean

If Dremio skipped the first line in the file or folder when creating the table, the value is `true`. Otherwise, the value is `false`.

Example: false

autoGenerateColumnNames Boolean

If Dremio used the existing column names in the file or folder for the table columns, the value is `true`. Otherwise, the value is `false`.

Example: true

trimHeader Boolean

If Dremio trimmed column names to a specific number of characters when creating the table, the value is `true`. Otherwise, the value is `false`.

Example: true

autoCorrectCorruptDates Boolean

If Dremio automatically corrects corrupted date fields in the table, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]]

roles Array of Object

Enterprise only. List of roles whose members have access to the table and the specific privileges each role has.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT", "ALTER"]}]]

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the table's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the table.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the table field.

Example: review_count

type Object

Information about the table field.

Attributes of the `type` Object

name String

Name of the table field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: BIGINT

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 0

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema appears only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: Parking

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Attributes of the `parquetDataFormat` Object

type String

Type of data in the table. Within the `parquetDataFormat` object, the only valid type is ``Parquet``.

Example: Parquet

ctime Integer

Not used. Has the value ``0``.

Example: 0

isFolder Boolean

If the value is ``true``, the table was created from a folder. If the value is ``false``, the table was created from a file.

Example: true

autoCorrectCorruptDates Boolean

If the value is ``true``, Dremio automatically corrects corrupted date fields in the table. Otherwise, the value is ``false``.

Example: true

Formatting a File or Folder as a Table

Format a file or folder as a table so that you can query the data in Dremio.

note

To format a folder, all files in the folder must be the same format.

Method and URL

```
POST /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the file or folder you want to format. The ID can be a UUID or a text path. If the ID is a text path, use URL encoding to replace special characters with their UTF-8-equivalent characters: ``%3A`` for a colon; ``%2F`` for a forward slash, and ``%20`` for a space. For example, if the ID value is ``dremio:/Samples/samples.dremio.com/Dremio University``, the URI-encoded ID is ``dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University``.

Example: c590ed7f-7142-4e1f-ba7d-94173afdc9a3

entityType Body String

Type of the catalog object. To format a file or folder as a table, the `entityType` is

``dataset``.

`path` Body Array of String

Path of the file or folder you want to format, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the file or folder itself as the last item in the array. Get the path from the file or folder's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

`type` Body String

Type of dataset. For tables, the type is ``PHYSICAL_DATASET``.

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

`format` Body String

Parameters that describe how to format the file or folder.

[accessControlList](#) Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

Parameters of the ``accelerationRefreshPolicy`` Object

`activePolicyType` Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the `neverRefresh` parameter must be set to ``false``.

The possible values are:

``NEVER``: The Reflections are never refreshed.

``PERIOD``: Default. The Reflections are refreshed at the end of every period that is defined by `refreshPeriodMs`.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in

Iceberg format.

refreshPeriodMs Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set activePolicyType to `PERIOD`. The default setting is `3600000` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

refreshSchedule Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set activePolicyType to `SCHEDULE`. The default refreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
-------	----------------	----------------------------

---	---	---
-----	-----	-----

Second	0	N/A
--------	---	-----

Minute	0-59	N/A
--------	------	-----

Hour	0-23	N/A
------	------	-----

Day of month	N/A	* ?
--------------	-----	------

Month	N/A	* ?
-------	-----	------

Days of week	1-7 or SUN-SAT	, - * ?
--------------	----------------	----------

Special Character	Description
-------------------	-------------

---	---
-----	-----

*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
----	--

?	Equivalent to *.
---	-------------------

,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
---	---

-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.
---	--

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the file or folder. All files in the folder must be the same format.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

ignoreOtherFileFormats Body Boolean Optional

If Dremio should ignore all non-Parquet files in the related folder structure so that the promoted table works as if only Parquet files are in the folder structure, set to `true`. Otherwise, set to `false` (default). Optional for Parquet folders.

Example: false

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the file or folder when creating the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for files or folders of the Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to ``true``. Otherwise, set to ``false`` (default). Optional for files or folders of the Excel and Text types.

Example: "

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the file when creating the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for files or folders of the Excel type.

Example: true

sheetName Body String Optional

For tables created from Excel files that contain multiple sheets, the name of the sheet to use to create the table. Default is the first sheet in the file (for files that contain multiple sheets).

Example: location_1

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be ```,`` for a comma (default), ``\t`` for a tab, ``|`` for a pipe, or a custom character. Optional for files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be ``"``` for a double quote (default), ``'`` for a single quote, or a custom character. Optional for files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments in the table. May be ``#`` for a number sign (default) or a custom character. Optional for files or folders of the Text type.

escape Body String Optional

Character used to indicate an escape in the table. May be ``"``` for a double quote (default), ``\`` for a back quote, ``\`` for a backward slash, or a custom character. Optional for files or folders of the Text type.

lineDelimiter Body String Optional

Character used to indicate separate lines in the table. May be ``\r\n`` for a carriage return and a new line (default), ``\n`` for a new line, or a custom character. Optional for files or folders of the Text type.

autoGenerateColumnNames Body Boolean Optional

If Dremio should use the existing column names in the file or folder for the table columns, set to `true` (default). Otherwise, set to `false`. Optional for files or folders of the Text type.

Example: true

trimHeader Body Boolean Optional

If Dremio should trim column names to a specific number of characters when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT","ALTER"]},{ "id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT","ALTER","MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
```

```
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Samples",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Parquet"
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ]
}
```

```

],
"createdAt": "2024-01-13T19:52:01.894Z",
"tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
"accelerationRefreshPolicy": {
  "activePolicyType": "PERIOD",
  "refreshPeriodMs": 3600000,
  "refreshSchedule": "0 56 18 * * *",
  "gracePeriodMs": 259200000,
  "method": "FULL",
  "neverExpire": true,
  "neverRefresh": false,
  "sourceRefreshOnDataChanges": false
},
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"format": {
  "type": "Parquet",
  "name": "restaurant_reviews.parquet",
  "fullPath": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "ctime": 0,
  "isFolder": false,
  "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
  "ignoreOtherFileFormats": "false",
  "autoCorrectCorruptDates": true
},
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
}

```

```

    ]
  }
]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",

```

```

    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "attributes",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "Parking",
          "type": {
            "name": "STRUCT",
            "subSchema": [
              {
                "name": "garage",
                "type": {
                  "name": "BOOLEAN"
                }
              },
              {
                "name": "street",
                "type": {
                  "name": "BOOLEAN"
                }
              },
              {
                "name": "lot",
                "type": {
                  "name": "BOOLEAN"
                }
              },
              {
                "name": "valet",
                "type": {
                  "name": "BOOLEAN"
                }
              }
            ]
          }
        }
      ],
      "name": "Accepts Credit Cards",
      "type": {
        "name": "BOOLEAN"
      }
    },
    {
      "name": "Wheelchair Accessible",
      "type": {
        "name": "BOOLEAN"
      }
    }
  },
  {

```

```

        "name": "Price Range",
        "type": {
            "name": "BIGINT"
        }
    }
]
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],
"approximateStatisticsAllowed": false
}

```

Example Request for Excel format type

```

curl -X POST 'https://{hostname}/api/v3/catalog/dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University%2Foracle-departments.xlsx' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
    "entityType": "dataset",
    "path": [
        "Samples",
        "samples.dremio.com",
        "Dremio University",
        "oracle-departments.xlsx"
    ],
    "type": "PHYSICAL_DATASET",
    "format": {
        "type": "Excel",
        "extractHeader": true,
        "hasMergedCells": true,
        "sheetName": "Sheet1"
    }
}'

```

Example Request for Text format type

```

curl -X POST 'https://{hostname}/api/v3/catalog/6ba3bd6e-fd27-4572-a535-77e1548283b3' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
    "entityType": "dataset",
    "path": [
        "Samples",

```

```
"samples.dremio.com",
"Dremio University",
"airbnb_listings.csv"
],
"type": "PHYSICAL_DATASET",
"format": {
  "type": "Text",
  "fieldDelimiter": ",",
  "skipFirstLine": false,
  "extractHeader": true,
  "quote": "\"",
  "comment": "#",
  "escape": "\\",
  "lineDelimiter": "\r\n",
  "autoGenerateColumnNames": true,
  "trimHeader": false
}
}'
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a Table by ID

Retrieve a table by specifying the table's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to retrieve.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
```

```

"users": [
  {
    "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
    "permissions": [
      "SELECT",
      "ALTER"
    ]
  },
  {
    "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "permissions": [
      "SELECT",
      "ALTER",
      "MANAGE_GRANTS"
    ]
  }
],
"roles": [
  {
    "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
    "permissions": [
      "SELECT",
      "ALTER"
    ]
  }
],
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  }
]

```

```

    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  }
},
{
  "name": "review_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "stars",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ],
        }
      },
      {
        "name": "street",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ],
  },
  {
    "name": "lot",
    "type": {
      "name": "BOOLEAN"
    }
  }
}

```

```

        },
        {
            "name": "valet",
            "type": {
                "name": "BOOLEAN"
            }
        }
    ]
}
},
{
    "name": "Accepts Credit Cards",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Wheelchair Accessible",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Price Range",
    "type": {
        "name": "BIGINT"
    }
}
]
}
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Retrieving a Table by Path

Retrieve a table by specifying the table's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Table's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" table in the "samples.dremio.com" folder within the source "Samples," the path is `Samples/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example:

Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
```

```

    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
},

```

```

"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  }
],

```

```

{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "street",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "lot",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "valet",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ]
        }
      },
      {
        "name": "Accepts Credit Cards",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "Wheelchair Accessible",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "Price Range",
        "type": {
          "name": "BIGINT"
        }
      }
    ]
  }
}

```



```

    }
  ]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Updating a Table

Update the specified table in Dremio.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

entityType Body String

Type of the catalog object. For tables, the entityType is `dataset`.

id Body String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

path Body Array of String

Path of the table that you want to update, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array. Get the path from the table's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

tag Body String (UUID) Optional

Unique identifier of the version of the table that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the table. If you do not provide a tag, Dremio automatically updates the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

type Body String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example:

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

[format](#) Body String

Parameters that describe the table's format.

[accessControlList](#) Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

Parameters of the `accelerationRefreshPolicy` Object

activePolicyType Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the neverRefresh parameter must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by refreshPeriodMs.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in Iceberg format.

`refreshPeriodMs` Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set `activePolicyType` to ``PERIOD``. The default setting is ``3600000`` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

`refreshSchedule` Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set `activePolicyType` to ``SCHEDULE``. The default `refreshSchedule` setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
-------	----------------	----------------------------

---	---	---
-----	-----	-----

Second	0	N/A
--------	---	-----

Minute	0-59	N/A
--------	------	-----

Hour	0-23	N/A
------	------	-----

Day of month	N/A	* ?
--------------	-----	------

Month	N/A	* ?
-------	-----	------

Days of week	1-7 or SUN-SAT	, - * ?
--------------	----------------	----------

Special Character	Description
-------------------	-------------

---	---
-----	-----

*	Used to specify all values for a field. For <code>`Day of month`</code> , specifies every day of the month. For <code>`Month`</code> , specifies every month. For <code>`Days of week`</code> , specifies every day of the week.
----	--

?	Equivalent to *.
---	-------------------

,	Used to specify two or more days in the <code>`Days of week`</code> field. For example, <code>`MON,WED,FRI`</code> .
---	--

-	Used to specify ranges in the <code>`Days of week`</code> field. For example, <code>`1-3`</code> is equivalent to <code>`Sunday, Monday, and Tuesday`</code> .
---	--

Examples:

``0 0 0 * * ?`` : Refreshes every day at midnight.

``0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

``0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

`gracePeriodMs` Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to ``true``. Otherwise, set to ``false`` (default). Optional for tables created from files or folders of the Excel and Text types.

Example: true

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the table, set to ``true``. Otherwise, set to ``false`` (default). Optional for tables created from files or folders of the Excel type.

Example: true

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be ``,`` for a comma (default), ``\t`` for a tab, ``|`` for a pipe, or a custom character. Optional for tables created from files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be ``\"`` for a double quote (default), ``'`` for a single quote, or a custom character. Optional for tables created from files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments for the table. May be ``#`` for a number sign (default) or a custom character. Optional for tables created from files or folders of the Text type.

escape Body String Optional

Character to use to indicate an escape for the table. May be ``\"`` for a double quote (default), ``\`` for a back quote, ``\\`` for a backward slash, or a custom character. Optional for tables created from files or folders of the Text type.

lineDelimiter Body String Optional

Character to use to indicate separate lines for the table. May be ``\r\n`` for a carriage return and a new line (default), ``\n`` for a new line, or a custom character. Optional for tables created from files or folders of the Text type.

Example:

autoGenerateColumnNames Body Boolean Optional

If Dremio should use the existing column names for the table columns, set to ``true`` (default). Otherwise, set to ``false``. Optional for tables created from files or folders of

the Text type.

Example: true

`trimHeader` Body Boolean Optional

If Dremio should trim column names to a specific number of characters when updating the table, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

`users` Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example:

```
[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ Grants"]}]
```

`roles` Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example:

```
[{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]
```

Parameters of Objects in the `users` and `roles` Arrays

`id` Body String

Enterprise only. Unique identifier of the user or role that should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

`permissions` Body Array of String

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/dba1e4fe-6351-44d2-a3e0-7aa20e782bf3' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "dba1e4fe-6351-44d2-a3e0-7aa20e782bf3",
  "path": [
```

```

    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Text",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  }
}'

```

Example Response

```

{
  "entityType": "dataset",
  "id": "dba1e4fe-6351-44d2-a3e0-7aa20e782bf3",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "createdAt": "2024-01-23T21:26:59.568Z",
  "tag": "fc1707df-35a1-45c1-87d7-5f66fb11a729",
  "format": {
    "type": "Text",
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/airbnb_listings.csv",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  },
  "accessControlList": {},
  "owner": {
    "ownerId": "c590ed7f-7142-4e1f-ba7d-94173afdc9a3",
    "ownerType": "USER"
  }
}

```

```

},
"fields": [
  {
    "name": "id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "listing_url",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "scrape_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "last_scraped",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "summary",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "reviews_per_month",
    "type": {
      "name": "VARCHAR"
    }
  }
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 500 Internal Server Error

Refreshing the Reflections on a Table

Refresh the Reflections associated with the specified table.

note

Refreshing a table's Reflections does not refresh its metadata. Read [Refreshing Metadata](#) to learn how to refresh table metadata. Read [Refreshing Reflections](#) for more information about refreshing Reflections.

Method and URL

```
POST /api/v3/catalog/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

```
No response
```

Response Status Codes

- 204 No Content
- 400 Bad Request
- 401 Unauthorized

403 Forbidden

404 Not Found

Reverting a Table to a File or Folder

Revert a table in a source to change the data in the table back to its original format, file or folder. For more information, read [Formatting Data to a Table](#) and [Removing Formatting on Data](#)

note

If a table is saved in your home space, the revert request will delete the table entirely. The revert endpoint only changes a table back to a file or folder if the table is saved in a source.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to revert to a file or folder.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

File](/25.x/reference/api/catalog/file)[Next

User-Defined Function](/25.x/reference/api/catalog/user-defined-function)

Table Attributes

Formatting a File or Folder as a Table

Parameters

Response Status Codes

Retrieving a Table by ID

Parameters

Response Status Codes

Retrieving a Table by Path

Parameters

Response Status Codes

Updating a Table

Parameters

Response Status Codes

Refreshing the Reflections on a Table

Parameters

Response Status Codes

Reverting a Table to a File or Folder

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/view/>

Version: 25.x

On this page

View

Use the Catalog API to retrieve, create, update, and delete views.

View Object

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
  ]
}
```

```

"VIEW_REFLECTION",
"MODIFY",
"MANAGE_GRANTS",
"CREATE_TABLE",
"DROP",
"EXTERNAL_QUERY",
"INSERT",
"TRUNCATE",
"DELETE",
"UPDATE",
"EXECUTE",
"CREATE_SOURCE",
"ALL"
],
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },

```

```

        {
            "name": "other",
            "type": {
                "name": "BOOLEAN"
            }
        }
    ]
}
},
{
    "name": "trip_distance_mi",
    "type": {
        "name": "DOUBLE"
    }
},
{
    "name": "fare_amount",
    "type": {
        "name": "DOUBLE"
    }
},
{
    "name": "tip_amount",
    "type": {
        "name": "DOUBLE"
    }
},
{
    "name": "total_amount",
    "type": {
        "name": "DOUBLE"
    }
}
]
}

```

View Attributes

entityType String

Type of the catalog object. For views, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the view.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

type String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

Example: VIRTUAL_DATASET

path Array of String

Path of the view within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the view itself as the last item in the array.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

createdAt String

Date and time that the view was created, in UTC format.

Example: 2022-11-17T18:31:23.236Z

isMetadataExpired Boolean

If true, the metadata of the tables that the view is defined on needs to be refreshed. To refresh it, run the ALTER VIEW command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the view.

If NULL, metadata has never yet been collected for the tables that the view is defined on.

lastMetadataRefreshAt String

Date and time that the metadata of the tables that the view is defined on was last refreshed. In UTC format.

Example: 2024-01-31T09:50:01.012Z

tag String (UUID)

Unique identifier of the version of the view. Dremio changes the tag whenever the view changes and uses the tag to ensure that PUT requests apply to the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql String

SQL query used to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Array of String

Context for the SQL query used to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Object

Enterprise only. Information about users and roles with access to the view and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The

accessControlList array is empty if view-specific access control privileges are not set.

Example: `{"users": [{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"] }, { "id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ GRANTS"] }, { "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}`

permissions Array of String

Enterprise-only. List of the privileges that you have on the view. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

`["READ", "WRITE", "ALTER\ REFLECTION", "SELECT", "ALTER", "VIEW\ REFLECTION", "MODIFY", "MANAGE\ GRANTS", "CREATE\ TABLE", "DROP", "EXTERNAL\ QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE\ SOURCE", "ALL"]`

owner String

Information about the view's owner.

Example: `{"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8", "ownerType": "USER"}`

fields Array of Object

Attributes that represent the dataset schema.

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the view and the specific privileges each user has.

Example: `[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ GRANTS"]}]`

roles Array of Object

Enterprise only. List of roles whose members have access to the view and the specific privileges each role has.

Example: `[{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]`

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the view. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the view's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the view.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the view field.

Example: pickup_datetime

type Object

Information about the view field.

Attributes of the `type` Object

name String

Name of the view field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: TIMESTAMP

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 2

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema is listed only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: cash

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Creating a View

Create a view from a table in Dremio.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the view itself as the last item in the array. The name of the view cannot include the following special characters: `/`, `:`, `[`, or `]`. Views can only be created in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

sql Body String

SQL query to use to create the view.

Example: `SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC`

`sqlContext Body` Array of String

Context for the SQL query to use to create the view.

Example: `["Samples", "samples.dremio.com"]`

`accessControlList Body` Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: `{ "users": [{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"] }, { "id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"] }], "roles": [{ "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"] }] }`

Parameters of the `accessControlList` Object

`users [Body]` Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: `[{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"] }, { "id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"] }]`

`roles Body` Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: `[{ "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"] }]`

Parameters of Objects in the `users` and `roles` Arrays

`id Body` String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: `c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3`

`permissions Body` Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: `["SELECT", "ALTER"]`

Example Request

```

curl -X POST 'https://{hostname}/api/v3/catalog/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "type": "VIRTUAL_DATASET",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}'

```

Example Response

```

{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",

```

```

"path": [
  "Business",
  "Transportation",
  "NYC-taxi-trips-short-distance"
],
"createdAt": "2022-11-17T18:31:23.236Z",
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
"sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
"sqlContext": [
  "Samples",
  "samples.dremio.com"
],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  }
],

```

```

{
  "name": "passenger_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "passenger_payment_method",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "cash",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "credit-debit",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "payment-app",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "other",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  }
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
}

```

```

    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a View by ID

Retrieve a view by specifying the view's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to retrieve.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
```

```
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{  
  "entityType": "dataset",  
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",  
  "type": "VIRTUAL_DATASET",  
  "path": [  
    "Business",  
    "Transportation",  
    "NYC-taxi-trips-short-distance"  
  ],  
  "createdAt": "2022-11-17T18:31:23.236Z",  
  "isMetadataExpired": false,  
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",  
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",  
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY  
trip_distance_mi ASC",  
  "sqlContext": [  
    "Samples",  
    "samples.dremio.com"  
  ],  
  "accessControlList": {  
    "users": [  
      {  
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",  
        "permissions": [  
          "SELECT",  
          "ALTER"  
        ]  
      },  
      {  
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",  
        "permissions": [  
          "SELECT",  
          "ALTER",  
          "MANAGE_GRANTS"  
        ]  
      }  
    ],  
    "roles": [  
      {  
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",  
        "permissions": [  
          "SELECT",  
          "ALTER"  
        ]  
      }  
    ]  
  },  
  "owner": {
```



```

    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  },
  "fields": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "passenger_payment_method",
      "type": {
        "name": "STRUCT",
        "subSchema": [
          {
            "name": "cash",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "credit-debit",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "payment-app",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "other",
            "type": {
              "name": "BOOLEAN"
            }
          }
        ]
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    }
  ],

```

```

{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a View by Path

Retrieve a view by specifying the view's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

View's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" view in the "samples.dremio.com" folder within the space "Transportation," the path is `Transportation/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent

characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Business/Transportation/NYC-taxi-trips-short-distance

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Business/Transportation/NYC-taxi-trips-short-distance' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
```

```

        "ALTER",
        "MANAGE_GRANTS"
    ]
}
],
"roles": [
{
    "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
    "permissions": [
        "SELECT",
        "ALTER"
    ]
}
]
},
"owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
},
"fields": [
{
    "name": "pickup_datetime",
    "type": {
        "name": "TIMESTAMP"
    }
},
{
    "name": "passenger_count",
    "type": {
        "name": "BIGINT"
    }
},
{
    "name": "passenger_payment_method",
    "type": {
        "name": "STRUCT",
        "subSchema": [
            {
                "name": "cash",
                "type": {
                    "name": "BOOLEAN"
                }
            },
            {
                "name": "credit-debit",
                "type": {
                    "name": "BOOLEAN"
                }
            },
            {
                "name": "payment-app",
                "type": {
                    "name": "BOOLEAN"
                }
            }
        ]
    }
}
]

```

```

    },
    {
      "name": "other",
      "type": {
        "name": "BOOLEAN"
      }
    }
  ]
}
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
]
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Updating a View

Update a view in Dremio.

Parameters

id Path String (UUID)

Unique identifier of the view that you want to update.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the updated view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the name for the view itself as the last item in the array. Views can only be saved in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

tag Body String (UUID) Optional

Unique identifier of the version of the view that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the view. If you do not provide a tag, Dremio automatically updates the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql Body String

SQL query to use to update the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC

sqlContext Body Array of String

Context for the SQL query to use for the updated view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\\GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "type": "VIRTUAL_DATASET",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ]
}
```

```
}'
```

Example Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-by-distance"
  ],
  "createdAt": "2023-01-20T15:26:39.780Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "7cab1a42-8835-4d31-827b-fedee1ad38d1",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
```



```

    "ownerType": "USER"
  },
  "fields": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "passenger_payment_method",
      "type": {
        "name": "STRUCT",
        "subSchema": [
          {
            "name": "cash",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "credit-debit",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "payment-app",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "other",
            "type": {
              "name": "BOOLEAN"
            }
          }
        ]
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    }
  ],
  {

```

```

    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Refreshing the Reflections on a View

Refresh the Reflections associated with the specified view.

Read [Refreshing Reflections](#) to learn how refreshing works.

Method and URL

POST /api/v3/catalog/{id}/refresh

Parameters

id Path String (UUID)

Unique identifier for the view you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example Request

```
curl -X POST 'https://api.dremio.cloud//api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Deleting a View

Delete the specified view.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to delete.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
-H 'Authorization: Bearer <PersonalAccessToken>' \
-H 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[\[Previous](#)

User-Defined Function][</25.x/reference/api/catalog/user-defined-function>][Next

Lineage][</25.x/reference/api/catalog/lineage>)

[View Attributes](#)

[Creating a View](#)

[Parameters](#)

[Response Status Codes](#)

[Retrieving a View by ID](#)

[Parameters](#)

[Response Status Codes](#)

[Retrieving a View by Path](#)

[Parameters](#)

[Response Status Codes](#)

[Updating a View](#)

[Parameters](#)

[Response Status Codes](#)

[Refreshing the Reflections on a View](#)

[Parameters](#)

[Response Status Codes](#)

[Deleting a View](#)

[Parameters](#)

[Response Status Codes](#)

Source:

Version: 25.x

On this page

Lineage Enterprise

Use the Catalog API to retrieve lineage information about datasets (tables and views). The lineage object includes information about the dataset's sources, parent objects, and child objects.

Lineage Object

```
{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
      "createdAt": "2022-02-14T21:57:48.794Z"
    }
  ],
  "parents": [
    {
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zips.json"
      ],
      "tag": "MAntohVzwLw=",
      "type": "DATASET",
      "datasetType": "PROMOTED",
      "createdAt": "2023-01-18T18:49:09.669Z"
    }
  ],
  "children": [
    {
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
      "path": [
        "@dremio",
        "NYC_zip"
      ],
      "tag": "OWKrfpEKzW4=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T02:11:46.344Z"
    }
  ],
}
```

```

{
  "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
  "path": [
    "@dremio",
    "Chicago_zip"
  ],
  "tag": "gsaDW5h4GCs=",
  "type": "DATASET",
  "datasetType": "VIRTUAL",
  "createdAt": "2023-01-25T00:09:12.461Z"
}
]
}

```

Lineage Attributes

sources Array of Object

Information about the sources the dataset uses. Each object in the sources array represents one source.

Example: [{"id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2","path": ["Samples"],"tag": "lz1v71CeTQY=", "type": "CONTAINER","containerType": "SOURCE","createdAt": "2023-02-14T21:57:48.794Z"}]

parents Array of Object

Information about the parent objects for the dataset. Each object in the parents array represents one parent object. If a view represents a join of two other datasets, the parents array includes the two joined datasets. The parents array is empty if the dataset does not have parent objects.

Example: [{"id": "3419fa3a-b5b3-4438-b864-a27ec4e18752","path": ["Samples","samples.dremio.com","zips.json"],"tag": "MAntohVzwLw=", "type": "DATASET","datasetType": "PROMOTED","createdAt": "2023-01-18T18:49:09.669Z"}]

children Array of Object

Information about other catalog objects that reference the dataset. Each object in the children array represents one child object. The children array is empty if the dataset does not have child objects.

Example: [{"id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75","path": ["@dremio","NYC_zip"],"tag": "OWKrfpEKzW4=", "type": "DATASET","datasetType": "VIRTUAL","createdAt": "2023-01-25T02:11:46.344Z"}, {"id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0","path": ["@dremio","Chicago_zip"],"tag": "gsaDW5h4GCs=", "type": "DATASET","datasetType": "VIRTUAL","createdAt": "2023-01-25T00:09:12.461Z"}]

Attributes of Objects in the `sources` Array

id String (UUID)

Unique identifier of the source associated with the dataset.

Example: 21077e5d-fe6f-4a29-843f-58fa3acb17c2

path Array of String

Path of the source within Dremio, expressed as an array.

Example: ["Samples"]

tag String

Unique identifier of the version of the source. Dremio changes the tag whenever the source changes.

Example: lz1v71CeTQY=

type String

Type of source. For sources in lineage responses, the type is `CONTAINER`.

Example: CONTAINER

containerType String

Type of container for the source.

Enum: HOME, SOURCE

Example: SOURCE

createdAt String

Date and time that the source was created, in UTC format. Not included for sources with the containerType `HOME`.

Example: 2022-02-14T21:57:48.794Z

Attributes of Objects in the `parents` Array

id String (UUID)

Unique identifier of the parent object.

Example: 3419fa3a-b5b3-4438-b864-a27ec4e18752

path Array of String

Path of the parent object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the parent object itself as the last item in the array.

Example: ["Samples","samples.dremio.com","zips.json"]

tag String

Unique identifier of the version of the parent object. Dremio changes the tag whenever the parent object changes.

Example: MAntohVzwLw=

type String

Type of parent object. For parent objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the parent object. If the parent object is a table, `PROMOTED`. If the parent object is a view, `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: PROMOTED

createdAt String

Date and time that the parent object was created, in UTC format.

Example: 2023-01-18T18:49:09.669Z

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the child object.

Example: 170e211e-4235-4d8d-acb5-3d4dbfe99c75

path Array of String

Path of the child object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the child object itself as the last item in the array.

Example: ["@dremio","NYC_zip"]

tag String

Unique identifier of the version of the child object. Dremio changes the tag whenever the child object changes.

Example: OWKrfpEKzW4=

type String

Type of child object. For child objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the child object. For child objects in lineage responses, the datasetType is `VIRTUAL`.

Example: VIRTUAL

createdAt String

Date and time that the child object was created, in UTC format.

Example: 2023-01-25T02:11:46.344Z

Retrieving Lineage Information About a Dataset

Retrieve lineage information about the specified dataset.

Method and URL

```
GET /api/v3/catalog/{id}/graph
```

Parameters

id Path String (UUID)

Unique identifier of the dataset whose lineage you want to retrieve.

Example: d69b25a3-31c8-4d55-a7cc-dfee2290779b

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/d69b25a3-31c8-4d55-a7cc-dfee2290779b/graph' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
      "createdAt": "2022-02-14T21:57:48.794Z"
    }
  ],
  "parents": [
    {
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zips.json"
      ],
    }
  ],
}
```

```

    "tag": "MAntohVzwLw=",
    "type": "DATASET",
    "datasetType": "PROMOTED",
    "createdAt": "2023-01-18T18:49:09.669Z"
  },
  ],
  "children": [
    {
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
      "path": [
        "@dremio",
        "NYC_zip"
      ],
      "tag": "0WKrfpEKzW4=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T02:11:46.344Z"
    },
    {
      "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
      "path": [
        "@dremio",
        "Chicago_zip"
      ],
      "tag": "gsaDW5h4GCs=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T00:09:12.461Z"
    }
  ]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[\[Previous](#)

[View\]\(/25.x/reference/api/catalog/view\)\[Next](#)

[Tag\]\(/25.x/reference/api/catalog/tag\)](#)

Lineage Attributes

Retrieving Lineage Information About a Dataset

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/tag/>

Version: 25.x

On this page

Tag

Use the Catalog API to create, update, and retrieve tags.

Tag Object

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VJ3ijXH4m6k="
}
```

Tag Attributes

tags Array of String

List of tags that apply to the dataset.

Example: ["NYC","taxi","2023"]

version String

Unique identifier of the set of tags. Dremio changes the version whenever any of the tags change and uses the version value to ensure that updates apply to the most recent version of the set of tags.

Example: VJ3ijXH4m6k=

Creating Tags

Create one or more tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset for which you want to add tags.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. Tags are case-insensitive. Each tag can be listed only once for each dataset. Each tag can have a maximum of 128 characters. Tags cannot include the following special characters: `/`, `:`, `[`, or `.`.

Example: ["NYC","taxi","2023"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023"]
}'
```

Example Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving Tags

Retrieve the tags applied to the specified dataset.

Method and URL

```
GET /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example Request

```
curl -X GET \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Updating Tags

Update the tags for the specified dataset.

Method and URL

POST /api/v3/catalog/{dataset-id}/collaboration/tag

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. If you want to keep any of the existing tags, include them in the tags array. Tags are case-insensitive and must be distinct (in other words, list each tag only once for each dataset). Each tag may have a maximum of 128 characters. Tags cannot include the following special characters: `/`, `:`, `[`, or ` `.

Example: ["NYC","taxi","2023","archived"]

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are updating the most recent version of the tags.

Example: VM3ijXH4m6k=

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023", "archived"],
  "version": "VM3ijXH4m6k="
}'
```

Example Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023",
    "archived"
  ],
  "version": "yiZSE++9wiU="
}
```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Deleting Tags

Delete the tags for the specified dataset.

note

Deleting tags means sending an empty array to replace the existing tags with no tags. The tag object will still exist, but it will contain an empty `tags` array and no tags will appear for the dataset in the Dremio UI.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to remove.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

Empty array to represent deletion of all tags for the dataset.

Example: []

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are deleting tags from the most recent version.

Example: yiZSE++9wiU=

Example Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
```

```
--header 'Content-Type: application/json' \  
--data-raw '{  
  "tags": [],  
  "version": "yiZSE++9wiU="  
'
```

Example Response

```
{  
  "tags": [],  
  "version": "wuTAKuRcVas="  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[Previous

Lineage](/25.x/reference/api/catalog/lineage)[Next

Wiki](/25.x/reference/api/catalog/wiki)

Tag Attributes

Creating Tags

Parameters

Response Status Codes

Retrieving Tags

Parameters

Response Status Codes

Updating Tags

Parameters

Response Status Codes

Deleting Tags

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/wiki/>

Version: 25.x

On this page

Wiki

Use the Catalog API to create, update, and retrieve the wiki for a source, space, or dataset.

Wiki Object

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio. Here is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n### Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",
  "version": 4
}
```

Wiki Attributes

text String

Text displayed in the wiki, formatted with GitHub-flavored Markdown.

version Integer

Number for the most recent version of the wiki, starting with `0`. Dremio increments the value by 1 each time the wiki changes and uses the version value to ensure that updates apply to the most recent version of the wiki.

Example: 4

Creating a Wiki

Create a wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset for which you want to add the wiki.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki. Use [GitHub-flavored Markdown](#) for wiki formatting and ``\n`` for new lines and blank lines. Each wiki may have a maximum of 100,000 characters.

Example Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum."
}'
```

Example Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
```

```
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving a Wiki

Retrieve the wiki for the specified source, space, or dataset.

Method and URL

```
GET /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example Request

```
curl -X GET
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wi
ki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.\nHere is some text in bold. Here is some text in italics.\n\nHere is an example\nexcerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur\nadipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna\naliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted\nlist\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###\nHeading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second\nitem in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that\nincludes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an\nimage:\n\n\n\nUt\nenim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea\ncommodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum\ndolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in\nculpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Updating a Wiki

Update the wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki, formatted with GitHub-flavored Markdown.

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are updating the most recent version of the wiki.

Example: 0

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/lbcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is an update to the bulleted list:\n* An item in
a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted
list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered
list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere
is a sentence that includes an [external link to
https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}'
```

Example Response

```
{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is an update to the bulleted list:\n* An item in
a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted
list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered
list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere
is a sentence that includes an [external link to
https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 1
}
```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Deleting a Wiki

Delete the wiki for the specified source, space, or dataset.

note

Deleting the wiki entails sending an empty string to replace the existing wiki with no wiki. The wiki object will still exist, but it will contain an empty `text` value and no wiki will appear for the source, space, or dataset in the Dremio UI.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to delete.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Empty string to represent deletion of the wiki.

Example: ""

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are deleting the most recent version of the wiki.

Example: 1

Example Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
```

```
--header 'Content-Type: application/json' \  
--data-raw '{  
  "text": "",  
  "version": 1  
}'
```

Example Response

```
{  
  "text": "",  
  "version": 2  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[Previous

Tag]/(25.x/reference/api/catalog/tag)[Next

Privileges]/(25.x/reference/api/catalog/privileges)

Wiki Attributes

Creating a Wiki

Parameters

Response Status Codes

Retrieving a Wiki

Parameters

Response Status Codes

Updating a Wiki

Parameters

Response Status Codes

Deleting a Wiki

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/privileges/>

Version: 25.x

On this page

Privileges Enterprise

caution

The Catalog API Privileges endpoint is deprecated. We expect to remove it by July 2025.

In place of the Privileges endpoint, use the Catalog API [Grants](#) endpoint to retrieve privileges and grantees on specific catalog objects.

Use the Catalog API to retrieve information about available privileges on the different types of catalog objects.

Privileges Object

```
{
  "availablePrivileges": [
    {
      "grantType": "SPACE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
      ]
    },
    {
      "grantType": "SOURCE",
      "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
      ]
    },
    {
      "grantType": "MUTABLE_SOURCE",
```



```

    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "CREATE_TABLE",
        "DELETE",
        "DROP",
        "INSERT",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "TRUNCATE",
        "UPDATE",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "ARP_SOURCE",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "EXTERNAL_QUERY",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER_IN_MUTABLE_SOURCE",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "CREATE_TABLE",
        "DELETE",
        "DROP",
        "INSERT",
        "MANAGE_GRANTS",
        "SELECT",
        "TRUNCATE",
        "UPDATE",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{

```

```

    "grantType": "PDS",
    "privileges": [
      "ALTER",
      "DELETE",
      "INSERT",
      "MANAGE_GRANTS",
      "SELECT",
      "TRUNCATE",
      "UPDATE"
    ]
  },
  {
    "grantType": "VDS",
    "privileges": [
      "ALTER",
      "MANAGE_GRANTS",
      "SELECT"
    ]
  },
  {
    "grantType": "FUNCTION",
    "privileges": [
      "ALTER",
      "EXECUTE",
      "MANAGE_GRANTS",
      "MODIFY"
    ]
  }
]
}
}

```

Privileges Attributes

availablePrivileges Array of Object

Information about the grant types and privileges that are available to assign to users and roles for each type of object in the catalog. Each availablePrivileges object contains two attributes: grantType and privileges.

Example:

```

[{"grantType": "SPACE", "privileges": ["ALTER", "ALTER\_REFLECTION", "MANAGE\_GRANTS", "MODIFY", "SELECT", "VIEW\_REFLECTION"]}, {"grantType": "SOURCE", "privileges": ["ALTER", "ALTER\_REFLECTION", "MANAGE\_GRANTS", "MODIFY", "SELECT", "VIEW\_REFLECTION"]}, {"grantType": "MUTABLE\ SOURCE", "privileges": ["ALTER", "ALTER\_REFLECTION", "CREATE\ _TABLE", "DELETE", "DROP", "INSERT", "MANAGE\ _GRANTS", "MODIFY", "SELECT", "TRUNCATE", "UPDATE", "VIEW\ _REFLECTION"]}, {"grantType": "ARP\ SOURCE", "privileges": ["ALTER", "ALTER\_REFLECTION", "EXTERNAL\ _QUERY", "MANAGE\ _GRANTS", "MODIFY", "SELECT", "VIEW\ _REFLECTION"]}, {"grantType": "FOLDER\ IN\ MUTABLE\ SOURCE", "privileges": ["ALTER", "ALTER\_REFLECTION", "CREATE\ _TABLE", "DELETE", "DROP", "INSERT", "MANAGE\ _GRANTS", "SELECT", "TRUNCATE", "UPDATE", "VIEW\ _REFLECTION"]}, {"grantType":

```

```
"FOLDER","privileges":
["ALTER","ALTER\_REFLECTION","MANAGE\_GRANTS","SELECT","VIEW\_REFLECTION"]},
{"grantType":
"PDS","privileges":
["ALTER","DELETE","INSERT","MANAGE\_GRANTS","SELECT","TRUNCATE","UPDATE"]},{"
grantType":
"VDS","privileges":
["ALTER","MANAGE\_GRANTS","SELECT"]},{"grantType":
"FUNCTION","privileges":
["ALTER","EXECUTE","MANAGE\_GRANTS","MODIFY"]}]
```

Attributes of Objects in the `availablePrivileges` Array

grantType String

Type of the catalog object on which the listed privileges are available. `ARP_SOURCE` refers to relational-database sources.

Enum: SPACE, SOURCE, MUTABLE_SOURCE, ARP_SOURCE, FOLDER_IN_MUTABLE_SOURCE, FOLDER, PDS, VDS, FUNCTION

Example: SPACE

privileges Array of String

List of available privileges on the type of the catalog object specified in grantType. For more information, read [Privileges](#).

Example:

```
["ALTER","ALTER\_REFLECTION","MANAGE\_GRANTS","MODIFY","SELECT","VIEW\_REFLE
CTION"]
```

Retrieving All Catalog Privileges

Retrieve information about the available privileges on each type of object in the catalog.

caution

This endpoint is deprecated. We expect to remove it by July 2025.

In place of this endpoint, use the Catalog API [Grants](#) endpoint to retrieve privileges and grantees on specific catalog objects.

Method and URL

```
GET /api/v3/catalog/privileges
```

Parameters

type Query String Optional

Type of the catalog object whose available privileges you want to retrieve. For more information, read [type Query Parameter](#).

Enum: SPACE, SOURCE, MUTABLE_SOURCE, ARP_SOURCE,

FOLDER_IN_MUTABLE_SOURCE, FOLDER, PDS, VDS, FUNCTION

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/privileges' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{  
  "availablePrivileges": [  
    {  
      "grantType": "SPACE",  
      "privileges": [  
        "ALTER",  
        "ALTER_REFLECTION",  
        "MANAGE_GRANTS",  
        "MODIFY",  
        "SELECT",  
        "VIEW_REFLECTION"  
      ]  
    },  
    {  
      "grantType": "SOURCE",  
      "privileges": [  
        "ALTER",  
        "ALTER_REFLECTION",  
        "MANAGE_GRANTS",  
        "MODIFY",  
        "SELECT",  
        "VIEW_REFLECTION"  
      ]  
    },  
    {  
      "grantType": "MUTABLE_SOURCE",  
      "privileges": [  
        "ALTER",  
        "ALTER_REFLECTION",  
        "CREATE_TABLE",  
        "DELETE",  
        "DROP",  
        "INSERT",  
        "MANAGE_GRANTS",  
        "MODIFY",  
        "SELECT",  
        "TRUNCATE",  
        "UPDATE",  
        "VIEW_REFLECTION"  
      ]  
    },  
    {  
      "grantType": "ARP_SOURCE",  
      "privileges": [  

```

```

        "ALTER",
        "ALTER_REFLECTION",
        "EXTERNAL_QUERY",
        "MANAGE_GRANTS",
        "MODIFY",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER_IN_MUTABLE_SOURCE",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "CREATE_TABLE",
        "DELETE",
        "DROP",
        "INSERT",
        "MANAGE_GRANTS",
        "SELECT",
        "TRUNCATE",
        "UPDATE",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "FOLDER",
    "privileges": [
        "ALTER",
        "ALTER_REFLECTION",
        "MANAGE_GRANTS",
        "SELECT",
        "VIEW_REFLECTION"
    ]
},
{
    "grantType": "PDS",
    "privileges": [
        "ALTER",
        "DELETE",
        "INSERT",
        "MANAGE_GRANTS",
        "SELECT",
        "TRUNCATE",
        "UPDATE"
    ]
},
{
    "grantType": "VDS",
    "privileges": [
        "ALTER",
        "MANAGE_GRANTS",
        "SELECT"
    ]
}

```

```
    },
    {
      "grantType": "FUNCTION",
      "privileges": [
        "ALTER",
        "EXECUTE",
        "MANAGE_GRANTS",
        "MODIFY"
      ]
    }
  ]
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Wiki](/25.x/reference/api/catalog/wiki)[Next

Grants](/25.x/reference/api/catalog/grants)

Privileges Attributes

Retrieving All Catalog Privileges

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/catalog/grants/>

Version: 25.x

On this page

Grants Enterprise

Use the Catalog API to grant user and role privileges on specific catalog objects.

Grants Object

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
      "firstName": "Jean",
      "lastName": "Smith",
      "email": "jean_smith@example.com"
    },
    {
      "privileges": [
        "ALTER",
        "SELECT"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "name": "examplerole"
    }
  ]
}
```

Grants Attributes

id String

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

availablePrivileges Array of String

List of available privileges on the catalog object.

Example:

["ALTER","DELETE","INSERT","MANAGE_GRANTS","SELECT","TRUNCATE","UPDATE"]

grants Array of Object

Information about the privileges and grantees for the catalog object. If the grants array is empty, there are no explicit grants for the object.

note

An empty grants array does not mean no users have access to the object at all. For example, admin users implicitly have all privileges on all catalog objects, owners implicitly have all privileges on everything they own, and children objects inherit the grants for their parent objects.

Example: [{"privileges": ["ALTER","SELECT","MANAGE\ Grants"], "granteeType": "USER", "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0", "name": "jeansmith", "firstName": "Jean", "lastName": "Smith", "email": "jean.smith@example.com"}, {"privileges": ["ALTER","SELECT"], "granteeType": "ROLE", "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "name": "examplerole"}]

Attributes of Objects in the `grants` Array

privileges String

List of privileges granted to the user or role. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE\ Grants"]

granteeType String

Type of grantee.

Enum: USER, ROLE

Example: USER

id String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

name String

Name of the user or role.

Example: jeansmith

firstName String

For users, the user's first name. Not included for roles.

Example: Jean

lastName String

For users, the user's last name. Not included for roles.

Example: Smith

email String

For users, the user's email address. Not included for roles.

Example: `jean_smith@example.com`

Creating or Updating Privilege Grants on a Catalog Object

Create or update the privileges granted to users and roles on the specified catalog object.

note

You must have the MANAGE GRANTS privilege to create or update privilege grants on catalog objects.

Method and URL

```
PUT /api/v3/catalog/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the Dremio catalog object.

Example: `7f1c4660-cd7b-40d0-97d1-b8a6f431cbda`

grants Body Array of Object

Array of objects that specify which users and roles should have privileges on the catalog object, as well as each user's and role's specific privileges. May include objects for users, roles, or both.

Example:

```
[{"privileges": ["ALTER","SELECT","MANAGE\_GRANTS"],"granteeType": "USER","id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"}, {"privileges": ["SELECT","ALTER"],"granteeType": "ROLE","id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"}]
```

Parameters of Objects in the `grants` Array

privileges Body Array of String

List of privileges to grant to the user or role. Use the Privileges endpoint to retrieve a list of available privileges on the catalog object type. For more information, read Privileges.

Example: `["ALTER","SELECT","MANAGE_GRANTS"]`

granteeType Body String

Type of grantee.

Enum: USER, ROLE

Example: USER

id Body String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"
    },
    {
      "privileges": [
        "SELECT",
        "ALTER"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"
    }
  ]
}'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieving Privileges and Grantees on a Catalog Object

Retrieve information about the privileges granted to users and roles on the specified catalog object.

note

Use this endpoint in place of the Catalog API Privileges endpoint, which is deprecated. We expect to remove the Privileges endpoint by July 2025.

You must have the MANAGE GRANTS privilege to retrieve privilege grants on catalog objects.

Method and URL

```
GET /api/v3/catalog/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the object whose privilege grants you want to retrieve.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

Example Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
```

```

    "firstName": "Jean",
    "lastName": "Smith",
    "email": "jean_smith@example.com"
  },
  {
    "privileges": [
      "ALTER",
      "SELECT"
    ],
    "granteeType": "ROLE",
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "name": "examplerole"
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Privileges](/25.x/reference/api/catalog/privileges)[Next

Dataset](/25.x/reference/api/datasets/)

Grants Attributes

Creating or Updating Privilege Grants on a Catalog Object

Parameters

Response Status Codes

Retrieving Privileges and Grantees on a Catalog Object

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/job/job-results>

Version: 25.x

On this page

Job Results

Use the Job API to retrieve results for a specific job.

Job Results Object

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "fare_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ],
  "rows": [
    {
      "pickup_datetime": "2013-02-10 20:00:00.000",
      "passenger_count": 9,
```

```

    "trip_distance_mi": 3,
    "fare_amount": 10,
    "tip_amount": 3,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:15:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.26,
    "fare_amount": 7.5,
    "tip_amount": 0,
    "total_amount": 8
  },
  {
    "pickup_datetime": "2013-05-31 16:40:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.73,
    "fare_amount": 5,
    "tip_amount": 1.2,
    "total_amount": 7.7
  },
  {
    "pickup_datetime": "2013-05-27 19:03:00.000",
    "passenger_count": 2,
    "trip_distance_mi": 9.23,
    "fare_amount": 27.5,
    "tip_amount": 5,
    "total_amount": 38.33
  },
  {
    "pickup_datetime": "2013-05-31 16:24:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.27,
    "fare_amount": 12,
    "tip_amount": 0,
    "total_amount": 13.5
  },
  {
    "pickup_datetime": "2013-05-27 19:17:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.71,
    "fare_amount": 5,
    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,
    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {

```

```

    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

Job Results Attributes

rowCount Integer

Number of rows the job returned.

Example: 11

schema Array of Object

Array of schema definitions for the data the job returned.

rows Array of Object

Array of the data the job returned for each row of results.

Retrieving Job Results

Method and URL

```
GET /api/v3/job/{id}/results
```

Parameters

id Path String (UUID)

Unique identifier of the job to retrieve. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

limit Query Integer Optional

Number of rows to retrieve. Maximum valid value is `500`. Default is `100`. Use with the `offset` query parameter to paginate and retrieve more than the first 500 results. Read [Limit and Offset Query Parameters](#) for usage examples.

offset Query Integer Optional

Number of rows to skip for pagination. Default is `0`. Read [Limit and Offset Query Parameters](#) for usage examples.

Example Request

```
curl -X GET 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00/results' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response for a COMPLETED Job

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    }
  ]
}
```



```

    },
    {
      "name": "fare_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ],
  "rows": [
    {
      "pickup_datetime": "2013-02-10 20:00:00.000",
      "passenger_count": 9,
      "trip_distance_mi": 3,
      "fare_amount": 10,
      "tip_amount": 3,
      "total_amount": 13
    },
    {
      "pickup_datetime": "2013-05-27 19:15:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 1.26,
      "fare_amount": 7.5,
      "tip_amount": 0,
      "total_amount": 8
    },
    {
      "pickup_datetime": "2013-05-31 16:40:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 0.73,
      "fare_amount": 5,
      "tip_amount": 1.2,
      "total_amount": 7.7
    },
    {
      "pickup_datetime": "2013-05-27 19:03:00.000",
      "passenger_count": 2,
      "trip_distance_mi": 9.23,
      "fare_amount": 27.5,
      "tip_amount": 5,
      "total_amount": 38.33
    }
  ]
}

```

```

    "pickup_datetime": "2013-05-31 16:24:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.27,
    "fare_amount": 12,
    "tip_amount": 0,
    "total_amount": 13.5
  },
  {
    "pickup_datetime": "2013-05-27 19:17:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.71,
    "fare_amount": 5,
    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,
    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {
    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }

```

```
}  
]  
}
```

For jobs with a `jobState` value other than ``COMPLETED``, the response includes an error message instead of the job results object. The job's status is listed inside brackets in the error message (``CANCELED``, ``FAILED``, or ``RUNNING``):

Example Response for a FAILED Job

```
{  
  "errorMessage": "Can not fetch details for a job that is in [FAILED] state.",  
  "moreInfo": ""  
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Job](/25.x/reference/api/job/)[Next

LDAP User Cache](/25.x/reference/api/ldap-authorization)

Job Results Attributes

Retrieving Job Results

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/job/job-results/>

Version: 25.x

On this page

Job Results

Use the Job API to retrieve results for a specific job.

Job Results Object

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "fare_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ],
  "rows": [
    {
      "pickup_datetime": "2013-02-10 20:00:00.000",
      "passenger_count": 9,
      "trip_distance_mi": 3,
      "fare_amount": 10,
      "tip_amount": 3,
      "total_amount": 13
    },
    {
      "pickup_datetime": "2013-05-27 19:15:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 1.26,
```

```

    "fare_amount": 7.5,
    "tip_amount": 0,
    "total_amount": 8
  },
  {
    "pickup_datetime": "2013-05-31 16:40:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.73,
    "fare_amount": 5,
    "tip_amount": 1.2,
    "total_amount": 7.7
  },
  {
    "pickup_datetime": "2013-05-27 19:03:00.000",
    "passenger_count": 2,
    "trip_distance_mi": 9.23,
    "fare_amount": 27.5,
    "tip_amount": 5,
    "total_amount": 38.33
  },
  {
    "pickup_datetime": "2013-05-31 16:24:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.27,
    "fare_amount": 12,
    "tip_amount": 0,
    "total_amount": 13.5
  },
  {
    "pickup_datetime": "2013-05-27 19:17:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.71,
    "fare_amount": 5,
    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,
    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {
    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",

```

```

    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

Job Results Attributes

rowCount Integer

Number of rows the job returned.

Example: 11

schema Array of Object

Array of schema definitions for the data the job returned.

rows Array of Object

Array of the data the job returned for each row of results.

Retrieving Job Results

Method and URL

```
GET /api/v3/job/{id}/results
```

Parameters

id Path String (UUID)

Unique identifier of the job to retrieve. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

limit Query Integer Optional

Number of rows to retrieve. Maximum valid value is `500`. Default is `100`. Use with the `offset` query parameter to paginate and retrieve more than the first 500 results. Read [Limit and Offset Query Parameters](#) for usage examples.

offset Query Integer Optional

Number of rows to skip for pagination. Default is `0`. Read [Limit and Offset Query Parameters](#) for usage examples.

Example Request

```
curl -X GET 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00/results' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response for a COMPLETED Job

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "fare_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
```

```

    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
],
"rows": [
  {
    "pickup_datetime": "2013-02-10 20:00:00.000",
    "passenger_count": 9,
    "trip_distance_mi": 3,
    "fare_amount": 10,
    "tip_amount": 3,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:15:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.26,
    "fare_amount": 7.5,
    "tip_amount": 0,
    "total_amount": 8
  },
  {
    "pickup_datetime": "2013-05-31 16:40:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.73,
    "fare_amount": 5,
    "tip_amount": 1.2,
    "total_amount": 7.7
  },
  {
    "pickup_datetime": "2013-05-27 19:03:00.000",
    "passenger_count": 2,
    "trip_distance_mi": 9.23,
    "fare_amount": 27.5,
    "tip_amount": 5,
    "total_amount": 38.33
  },
  {
    "pickup_datetime": "2013-05-31 16:24:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.27,
    "fare_amount": 12,
    "tip_amount": 0,
    "total_amount": 13.5
  },
  {
    "pickup_datetime": "2013-05-27 19:17:00.000",
    "passenger_count": 1,

```



```

    "trip_distance_mi": 0.71,
    "fare_amount": 5,
    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,
    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {
    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

For jobs with a jobState value other than `COMPLETED`, the response includes an error message instead of the job results object. The job's status is listed inside brackets in the error message (`CANCELED`, `FAILED`, or `RUNNING`):

Example Response for a FAILED Job

```
{
  "errorMessage": "Can not fetch details for a job that is in [FAILED] state.",
  "moreInfo": ""
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Job](/25.x/reference/api/job/)[Next

LDAP User Cache](/25.x/reference/api/ldap-authorization)

Job Results Attributes

Retrieving Job Results

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/reflections/reflection-recommendations>

Version: 25.x

On this page

Recommendations

Use the Recommendations API to get job-based recommendations and get and create usage-based Reflections that can accelerate your queries.

Getting **job-based recommendations** requires making the following two API requests:

Submit the job IDs of jobs that have run SQL queries. These are job IDs of the queries for which you want to retrieve recommendations in further requests. This request returns the job ID to use in the second request.

Retrieve job-based recommendations for Reflections that can accelerate your queries. Use the job ID that was returned in your first request to make the request for recommendations.

Creating Reflections from **usage-based recommendations** requires making the following two API requests:

Retrieve usage-based recommendations for Reflections. This request returns the parameters to use in the body of the second request.

Create Reflections from usage-based recommendations that can accelerate your queries from the usage-based recommendations. Use the recommendation ID and Reflection request body that were returned in your first request to create the Reflections.

note

Dremio supports creating only raw Reflections from usage-based recommendations.

caution

The ``POST /api/v3/reflection/recommendations` endpoint` is deprecated. In its place, use the job-based and usage-based endpoints described on this page to retrieve and refresh Reflection recommendations.

Recommendation Object (Raw Reflection)

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "azure_3",
          "table_2"
        ],
        "sql": "--Default Raw Reflection"
      },
      "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
      "reflectionRequestBody": {
        "type": "RAW",
        "name": "raw_47f54460-543f-430f-a9e5-ca71d246265e",
        "datasetId": "45b9d98b-b0dc-4dd2-a271-d971ae998c0c",
        "enabled": true,
        "arrowCachingEnabled": false,
        "dimensionFields": [],
        "measureFields": [],
        "displayFields": [
          {
            "name": "passenger_count"
          },
          {
            "name": "EXPR$1"
          }
        ],
        "entityType": "reflection"
      },
      "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
      "jobIds": [
```

```

    "13ffb629-9f0e-4265-97df-99bf0d425813"
  ],
  "jobCount": 1,
  "recommendationId": "9be8a451-4190-4618-a72e-9932f790c744",
  "reflectionScore": 50.67,
  "avgImprovementFactor": 10.43,
  "avgImprovementMs": 7196
}
],
"canAlterReflections": true
}

```

Recommendation Object (Aggregation Reflection)

```

{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "recommended_view",
          "view_1"
        ],
        "sql": "SELECT * FROM Samples.samples.dremio.com.\"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC"
      },
      "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
      "reflectionRequestBody": {
        "type": "AGGREGATION",
        "name": "agg_0e0c4ab9-def7-48da-81f1-ca8c1dalled4",
        "datasetId": "2df93b5a-eb46-4687-8460-b61e471d20ef",
        "enabled": true,
        "arrowCachingEnabled": false,
        "dimensionFields": [
          {
            "name": "passenger_count",
            "granularity": "DATE"
          }
        ],
        "measureFields": [
          {
            "name": "fare_amount",
            "measureTypeList": [
              "SUM",
              "COUNT"
            ]
          }
        ],
        "displayFields": [],
        "entityType": "reflection"
      },
      "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
    }
  ]
}

```

```

    "jobIds": [
      "1ded81f8-4d06-4d09-8163-9e2517027d8d"
    ],
    "jobCount": 1,
    "recommendationId": "1855d2dd-4106-4359-a97a-e08a916096e6",
    "reflectionScore": 60.12,
    "avgImprovementFactor": 8.39,
    "avgImprovementMs": 5400
  }
],
"canAlterReflections": true
}

```

Recommendation Attributes

data Array of Object

List of recommended Reflection objects for the submitted job IDs.

canAlterReflections Boolean

If the columns in the recommended Reflection can be edited, added, and removed, `true`. Otherwise, `false`.

Example: true

Attributes of Objects in the `data` Array

viewRequestBody Object

The fields to include in a request to the Catalog API to create the view on which to define the recommended Reflection.

viewRequestEndpoint String

The endpoint to use when submitting a request to the Catalog API to create the view on which to define the recommended Reflection.

reflectionRequestBody Object

The fields to include in a request to the Reflection API to create the recommended Reflection.

reflectionRequestEndpoint String

The endpoint to use when submitting the request to the Reflection API to create the recommended Reflection.

jobIds Array of String

The job IDs of the queries for which the Reflection recommendations are given.

Example: ["13ffb629-9f0e-4265-97df-99bf0d425813"]

jobCount Array of String

The number of jobs for which Reflection recommendations are given.

Example: 1

recommendationId Array of String

The ID of the recommended Reflection.

Example: ["9be8a451-4190-4618-a72e-9932f790c744"]

reflectionScore Double

Score for the recommended Reflection's quality, on a scale of 0 (worst) to 100 (best). The reflectionScore value considers the recommended Reflection's anticipated quality compared to existing Reflections and other recommended Reflections, as well as the likely improvement in query run times if the recommended Reflection is implemented.

Example: 50.67

avgImprovementFactor Double

The likely average multiplicative rate of improvement for each query if you implement the recommended Reflection. For example, if the avgImprovementFactor value is 2.34, implementing the recommended Reflection is likely to speed up each query by 2.34 times, on average.

Example: 10.43

avgImprovementMs Double

The likely average improvement, in milliseconds, for each query if you implement the recommended Reflection. For example, if the avgImprovementMs value is 5400, implementing the recommended Reflection is likely to save an average of 5400 milliseconds for each query that uses the Reflection.

Example: 7196

Attributes of the `viewRequestBody` Object

entityType String

Type of catalog entity. For views, the entityType is `dataset`.

type String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

path Array of String

Path to the location where the view should be created within Dremio, expressed in an array. The path lists each level of hierarchy in order, from outer to inner: Arctic source or catalog first, then folder and subfolders, then a name for the view itself as the last item in the array. Views can only be created in Arctic sources and the project's Arctic catalog.

Example: ["azure_3","table_2"]

sql String

For aggregation Reflections, the SQL query to use to create the view. For default raw Reflections, the sql value ``Default Raw Reflection``; creating a view is unnecessary because raw recommendations are given only for existing views.

Attributes of the ``reflectionRequestBody`` Object

type String

Reflection type. For details, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio console, if the user did not provide a name, the default values are ``Raw Reflection`` and ``Aggregation Reflection`` (automatically assigned based on the Reflection type).

Example: raw_47f54460-543f-430f-a9e5-ca71d246265e

datasetId String

Unique identifier for the anchor dataset to associate with the Reflection.

Example: 45b9d98b-b0dc-4dd2-a271-d971ae998c0c

enabled Boolean

If the Reflection is available for accelerating queries, ``true``. Otherwise, ``false``.

Example: true

arrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, ``true``. Otherwise, ``false``.

Example: false

dimensionFields Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Valid only for aggregation Reflections. For raw Reflections or if the anchor dataset does not include any dimension fields, the dimensionFields value is an empty array. For aggregation Reflections, if the anchor dataset includes dimension fields, each object in the dimensionFields array contains two attributes: name and granularity.

Example: [{"name": "pickup_date","granularity": "DATE"}, {"name": "pickup_datetime","granularity": "DATE"}, {"name": "dropoff_date","granularity": "DATE"}, {"name": "dropoff_datetime","granularity": "DATE"}]

```
"passenger\_count","granularity": "DATE"}, {"name": "total\_amount","granularity": "DATE"}]
```

measureFields Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Valid only for aggregation Reflections. For raw Reflections or if the anchor dataset does not include any measure fields, the measureFields value is an empty array. For aggregation Reflections, if the anchor dataset includes measure fields, each object in the measureFields array contains two attributes: name and measureTypeList.

```
Example: [{"name": "passenger\_count","measureTypeList": ["SUM","COUNT"]}, {"name": "trip\_distance\_mi","measureTypeList": ["SUM","COUNT"]}, {"name": "fare\_amount","measureTypeList": ["SUM","COUNT"]}, {"name": "surcharge","measureTypeList": ["SUM","COUNT"]}, {"name": "tip\_amount","measureTypeList": ["SUM","COUNT"]}, {"name": "total\_amount","measureTypeList": ["SUM","COUNT"]}]
```

displayFields Array of Object

Information about the fields displayed from the anchor dataset. Valid only for raw Reflections. For aggregation Reflections or if the anchor dataset does not include any display fields, the value is an empty array. For raw Reflections, if the anchor dataset includes display fields, each object in the displayFields array contains one attribute: name.

```
Example: [{"name": "passenger\_count"}, {"name": "EXPR$1"}]
```

entityType String

Type of entity. For Reflection objects, the entityType is `reflection`.

Submitting Job IDs

Submit the job IDs of queries for which you want to request Reflection recommendations.

The response includes objects that contain an id attribute and value for each job ID you submit. Use these id values to [retrieve recommendations for Reflections](#) to accelerate the queries.

Method and URL

```
POST /api/v3/reflection/recommendations/job-based/
```

Parameters

jobIds Body Array of String

The job IDs of the queries for which you want to request Reflection recommendations. To get the job IDs, use the [SQL API](#) or find them on the [Jobs page](#) in the Dremio console. Use a comma-separated list to submit multiple job IDs.

Example: ["a7efcd50-791a-48e8-bb05-391b4411e66b"]

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations/job-based/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
    "jobIds":
    ["a7efcd50-791a-48e8-bb05-391b4411e66b", "c2485882-e6b7-4aa8-af5b-a825d2870589"]
}'
```

Example Response

```
{
  "id": "13ffb629-9f0e-4265-97df-99bf0d425813"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

405 Method Not Allowed

500 Internal Server Error

Retrieving Job-Based Recommendations

Retrieve job-based recommended Reflections to accelerate the queries whose job IDs you submitted.

For default raw Reflections, each recommendation comprises the path to the view on which to define the Reflection and the parameters to use in a request to create the Reflection.

For aggregation Reflections, each recommendation comprises the parameters to use in a request to create a view on which to define the recommended Reflection and the parameters to use in a request to create the Reflection.

After you retrieve the recommended Reflections for your queries, use the Catalog API to create the recommended views. Then, use the Reflection API to create the desired Reflections.

note

Before submitting Catalog API requests to create the recommended views for aggregation Reflections, create a folder named ``recommended_view``. In your Catalog API requests, the ``path`` parameter must include the full path to the ``recommended_view`` folder. If you prefer to use a different folder name, replace

`recommended_view` with your folder name in the `path` parameter when making the Catalog API request.

Method and URL

```
GET /api/v3/reflection/recommendations/job-based/{id}/results/
```

Parameters

id Path String

The id value returned in the response to your request to submit the job ID or IDs of the queries for which you want to retrieve recommended Reflections.

Example: 13ffb629-9f0e-4265-97df-99bf0d425813

Example Request

```
curl -X GET \
'https://{hostname}/api/v3/reflection/recommendations/job-based/13ffb629-9f0e-4265-97df-99bf0d425813/results/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "azure_3",
          "table_2"
        ],
        "sql": "--Default Raw Reflection"
      },
      "viewRequestEndpoint": "{hostname}/api/v3/catalog",
      "reflectionRequestBody": {
        "type": "RAW",
        "name": "raw_47f54460-543f-430f-a9e5-ca71d246265e",
        "datasetId": "45b9d98b-b0dc-4dd2-a271-d971ae998c0c",
        "enabled": true,
        "arrowCachingEnabled": false,
        "dimensionFields": [],
        "measureFields": [],
        "displayFields": [
          {
            "name": "passenger_count"
          },
          {
            "name": "EXPR$1"
          }
        ]
      }
    }
  ]
}
```

```

    }
  ],
  "entityType": "reflection"
},
"reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
"jobIds": [
  "13ffb629-9f0e-4265-97df-99bf0d425813"
],
"jobCount": 1,
"recommendationId": "9be8a451-4190-4618-a72e-9932f790c744",
"reflectionScore": 50.67,
"avgImprovementFactor": 10.43,
"avgImprovementMs": 7196
}
],
"canAlterReflections": true
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

405 Method Not Allowed

500 Internal Server Error

Retrieving Usage-Based Recommendations

Retrieve usage-based Reflection recommendations. The response includes the ``reflectionRequestBody`` and ``recommendationId`` attributes to use as body parameters in your request to [create usage-based Reflections](#).

Method and URL

```
GET /api/v3/reflection/recommendations/usage-based/
```

Example Request

```

curl -X GET 'https://{hostname}/api/v3/reflection/recommendations/usage-based/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'

```

Example Response

```

{
  "data": [
    {

```

```

"viewRequestBody": {
  "entityType": "dataset",
  "type": "VIRTUAL_DATASET",
  "path": [
    "prodFolder",
    "cost_based"
  ],
  "sql": "--Default Raw Reflection"
},
"viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
"reflectionRequestBody": {
  "type": "RAW",
  "name": "AutoRef_cost_based_raw",
  "datasetId": "61d689a2-cd04-4d5d-84a7-021bdc15bff6",
  "enabled": true,
  "arrowCachingEnabled": false,
  "dimensionFields": [],
  "measureFields": [],
  "displayFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime"
    },
    {
      "name": "EXPR$2"
    }
  ],
  "entityType": "reflection"
},
"reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
"jobIds": [
  "1975ec43-349a-9310-2e40-acbd8d025c00",
  "1975ac7c-6541-86db-ae43-dcef2ffee300",
  "1974b533-8c88-946b-92ce-ee5ab7791500"
],
"jobCount": 3,
"recommendationId": "prodFolder.cost_based",
"reflectionScore": 36.928031592652964,
"avgImprovementFactor": 10.00000020692081,
"avgImprovementMs": 7393.800016999235
}
],
"canAlterReflections": true
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

500 Internal Server Error

Creating Reflections from Usage-Based Recommendations

Create Reflections to accelerate queries using the usage-based recommendations that you retrieved.

note

You must retrieve usage-based recommendations to get the parameters you need for this request.

Dremio supports creating only raw Reflections from usage-based recommendations.

Method and URL

```
POST /api/v3/reflection/recommendations/usage-based/
```

Parameters

reflection Body Object

Information about the usage-based Reflection to create. The Reflection object includes the contents of the reflectionRequestBody included in the response for requests to retrieve usage-based recommendations.

NOTE: If desired, you may change the name of the Reflection by changing the value for the Reflection.name parameter in the body of your request. Dremio ignores any changes to the values of other parameters in the Reflection object.

recommendationId Body String

Identifier for the usage-based recommendation you want to use to create Reflections. The recommendationId is included in the response for requests to retrieve usage-based recommendations.

Example: prodFolder.cost_based

Parameters of the `reflection` Object

type Body String

Reflection type. Value must be `RAW`.

name Body String

User-provided name for the Reflection.

NOTE: If desired, you may change the name of the Reflection by changing the value for the name parameter in the body of your request.

Example: AutoRef_cost_based_raw

datasetId Body String

Unique identifier for the anchor dataset to associate with the Reflection.

Example: 61d689a2-cd04-4d5d-84a7-021bdc15bff6

enabled Body Boolean

If the Reflection is available for accelerating queries, `true`. Otherwise, `false`.

Example: true

arrowCachingEnabled Body Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, `true`. Otherwise, `false`.

Example: false

dimensionFields Body Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. For raw Reflections, the dimensionFields value is an empty array.

Example: []

measureFields Body Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. For raw Reflections, the measureFields value is an empty array.

Example: []

displayFields Body Array of Object

Information about the fields displayed from the anchor dataset. Valid only for raw Reflections. If the anchor dataset includes display fields, each object in the displayFields array contains one attribute: name.

Example: [{"name": "passenger_count"}, {"name": "EXPR\$1"}]

entityType Body String

Type of entity. For Reflection objects, the entityType is `reflection`.

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations/usage-based/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "reflection": {
    "type": "RAW",
    "name": "AutoRef_cost_based_raw",
    "datasetId": "61d689a2-cd04-4d5d-84a7-021bdc15bff6",
    "enabled": true,
    "arrowCachingEnabled": false,
```

```

"dimensionFields": [],
"measureFields": [],
"displayFields": [
  {
    "name": "passenger_count"
  },
  {
    "name": "pickup_datetime"
  },
  {
    "name": "EXPR$2"
  }
],
"entityType": "reflection"
},
"recommendationId": "prodFolder.cost_based"
}

```

Example Response

```

{
  "id": "c929b8d2-82bf-4175-9476-010ba17c4f7f",
  "type": "RAW",
  "name": "AutoRef_cost_based_raw",
  "tag": "4p/C0EkSud7=",
  "createdAt": "2024-06-28T19:30:30.977Z",
  "updatedAt": "2024-06-28T19:30:30.977Z",
  "datasetId": "61d689a2-cd04-4d5d-84a7-021bdc15bff6",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "NONE",
    "combinedStatus": "CANNOT_ACCELERATE_SCHEDULED",
    "failureCount": 0,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "displayFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime"
    },
    {
      "name": "EXPR$2"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",

```

```
"canView": true,  
"canAlter": true,  
"entityType": "reflection"  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

500 Internal Server Error

Refreshing Usage-Based Recommendations

Process collected data about view usage, clear existing usage-based recommendations, and generate new usage-based recommendations.

Use the usage-based endpoints to retrieve and create Reflections based on the refreshed recommendations this endpoint creates.

Method and URL

```
POST /api/v3/reflection/recommendations/usage-based/refresh/
```

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations/usage-based/refresh/' \  
\   
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

This endpoint returns an empty response body with a `202 Accepted` response status code. Dremio updates the recommendations asynchronously, so it may take several minutes before you can retrieve the updated recommendations.

Response Status Codes

202 Accepted

400 Bad Request

401 Unauthorized

500 Internal Server Error

Deleting Usage-Based Recommendations

Delete all collected usage data and all current Reflection recommendations.

note

We recommend deleting recommendations only when troubleshooting.

Method and URL

```
DELETE /api/v3/reflection/recommendations/usage-based/
```

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/reflection/recommendations/usage-based/' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

This endpoint returns an empty response body with a `202 Accepted` response status code. Dremio deletes the recommendations asynchronously, so it may take several minutes for the deletion to complete.

Response Status Codes

202 Accepted

400 Bad Request

401 Unauthorized

500 Internal Server Error

Retrieving Recommendations (Deprecated)"

caution

The `POST /api/v3/reflection/recommendations` endpoint described in this section is deprecated. In its place, use the `job-based` and `usage-based` endpoints to retrieve and refresh Reflection recommendations.

Use the Recommendations API to submit job IDs of jobs that ran SQL queries, and receive recommendations for aggregation Reflections that can accelerate those queries.

For more information, see [Sending Requests to the Recommendations API](#).

Recommendation Object

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "path": [
          "recommended_view",
```

```

        "Dataset_be919a56-f18b-421b-9612-711a1cc51b69"
    ],
    "type": "VIRTUAL_DATASET",
    "sql": "SELECT * FROM Samples.samples.dremio.com.\"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
    }
    "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
    "reflectionRequestBody": {
        "type": "AGGREGATION",
        "name": "agg_250e70d1-5e2a-4938-a1a1-95f664085099",
        "datasetId": "be919a56-f18b-421b-9612-711a1cc51b69",
        "enabled": true,
        "dimensionFields": [
            {
                "name": "passenger_count",
                "granularity": "DATE"
            }
        ],
        "measureFields": [
            {
                "name": "fare_amount",
                "measureTypeList": [
                    "SUM",
                    "COUNT"
                ]
            }
        ],
        "entityType": "reflection"
    }
    "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
    "jobIds": ["6j6c34cf-9drf-b07a-5ab7-abea69a66d00"]
    }
    ],
    "canAlterReflections": true
}

```

Recommendation Attributes (Deprecated)"

viewRequestBody Array of Object

The fields that you can include in a request to the Catalog API for creating the view on which to define the recommended aggregation Reflection.

For descriptions of these fields, see [View](#).

viewRequestEndpoint String

The endpoint to use when submitting the request to the Catalog API to create the view on which to define the Reflection.

reflectionRequestBody Array of Object

The fields that you can include in a request to the Reflection API for creating the

recommended aggregation Reflection.

For descriptions of these fields, see [Reflection](#).

reflectionRequestEndpoint String

The endpoint to use when submitting the request to the Reflection API to create the aggregation Reflection.

jobIds Array of String

The IDs of the jobs that ran the queries for which the recommendation is given.

canAlterReflections String

Indicates whether the columns in the Reflection can be edited, and whether columns can be added or removed.

Requesting Recommendations (Deprecated)"

caution

This endpoint is deprecated. In its place, use the [job-based and usage-based](#) endpoints to retrieve and refresh Reflection recommendations.

Request recommended aggregation Reflections to accelerate the queries associated with the provided job ID or IDs.

Method and URL

```
POST /api/v3/reflection/recommendations
```

Parameters (Deprecated)"

jobIds Body Array of String

A list of the job IDs of jobs that have run the SQL commands that you want to receive one or more recommended Reflections for.

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data
'{
  "jobIds": [
    "1a515250-7572-0f9b-f5e5-89f505b55200",
    "1a515292-583c-e407-79ef-9f9b494fa600"
  ]
}'
```

Was this page helpful?

[Previous

Reflection](/25.x/reference/api/reflections/)[Next

Reflection Summary](/25.x/reference/api/reflections/reflection-summary)

Recommendation Attributes

Submitting Job IDs

Parameters

Response Status Codes

Retrieving Job-Based Recommendations

Parameters

Response Status Codes

Retrieving Usage-Based Recommendations

Response Status Codes

Creating Reflections from Usage-Based Recommendations

Parameters

Response Status Codes

Refreshing Usage-Based Recommendations

Response Status Codes

Deleting Usage-Based Recommendations

Response Status Codes

Retrieving Recommendations (Deprecated)

Source:
<https://docs.dremio.com/25.x/reference/api/reflections/reflection-summary>

Version: 25.x

On this page

Reflection Summary Enterprise

Use the Reflection API to retrieve a Reflection summary that includes the raw and aggregation Reflections for the Dremio instance.

Reflection summary objects are different from Reflection objects. Reflection summaries do not include certain attributes that define the Reflection, like the display, dimension, measure, sort, and partition attributes. Reflection summaries do include several attributes that do not appear in Reflection objects, like `datasetType`, `datasetPath`, and

counts and links for considered, matched, and chosen jobs.

Reflection Summary Object

```
{
  "data": [
    {
      "createdAt": "2022-07-05T19:19:40.244Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
      "totalSizeBytes": 9272,
      "datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
      "datasetType": "VIRTUAL_DATASET",
      "datasetPath": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "status": {
        "configStatus": "OK",
        "refreshStatus": "MANUAL",
        "availabilityStatus": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "refreshMethod": "FULL",
        "failureCount": 0,
        "lastDataFetchAt": "2023-01-13T19:05:03.532Z",
        "expiresAt": "3022-05-16T19:46:02.342Z",
        "lastRefreshDurationMillis": 46387
      },
      "consideredCount": 202,
      "matchedCount": 45,
      "chosenCount": 5,
      "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "isArrowCachingEnabled": false,
      "isCanView": true,
      "isCanAlter": true,
      "isEnabled": true
    },
    {
      "createdAt": "2023-01-26T23:27:04.281Z",
      "updatedAt": "2023-01-26T23:27:04.281Z",
      "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
      "reflectionType": "RAW",

```

```

"name": "Raw Reflection",
"currentSizeBytes": 0,
"outputRecords": -1,
"totalSizeBytes": 0,
"datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
"datasetType": "PHYSICAL_DATASET",
"datasetPath": [
  "@dremio",
  "test"
],
"status": {
  "configStatus": "OK",
  "refreshStatus": "GIVEN_UP",
  "availabilityStatus": "NONE",
  "combinedStatus": "FAILED",
  "refreshMethod": "NONE",
  "failureCount": 3,
  "lastFailureMessage": "The Default engine is not online.",
  "lastDataFetchAt": null,
  "expiresAt": null,
  "lastRefreshDurationMillis": -1
},
"consideredCount": 0,
"matchedCount": 0,
"chosenCount": 0,
"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"isArrowCachingEnabled": false,
"isCanView": true,
"isCanAlter": true,
"isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:19.030Z",
  "updatedAt": "2023-01-13T19:50:19.030Z",
  "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
  "reflectionType": "AGGREGATION",
  "name": "dataset991_agg991",
  "currentSizeBytes": 9273,
  "outputRecords": 51,
  "totalSizeBytes": 9273,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],

```

```

"status": {
  "configStatus": "OK",
  "refreshStatus": "MANUAL",
  "availabilityStatus": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "refreshMethod": "FULL",
  "failureCount": 0,
  "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
  "expiresAt": "3022-05-16T19:46:02.342Z",
  "lastRefreshDurationMillis": 11697
},
"consideredCount": 60,
"matchedCount": 9,
"chosenCount": 0,
"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"isArrowCachingEnabled": false,
"isCanView": true,
"isCanAlter": true,
"isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:17.714Z",
  "updatedAt": "2023-01-13T19:50:17.714Z",
  "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
  "reflectionType": "RAW",
  "name": "dataset991_raw991",
  "currentSizeBytes": 818790,
  "outputRecords": 29467,
  "totalSizeBytes": 818790,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
    "expiresAt": "3022-05-16T19:46:02.747Z",
    "lastRefreshDurationMillis": 16666
  },

```

```

    "consideredCount": 54,
    "matchedCount": 37,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  }
],
"nextPageToken": "CiQxNjc0MjhIYi03OTM2LTRlYtItYTFmYi0yM2IxYWM2ZTk0NTQSAA==",
"isCanAlterReflections": true
}

```

Reflection Summary Attributes

data Array of Object

List of Reflection-summary objects for each Reflection in the Dremio instance.

nextPageToken String

Opaque string to pass for the `pageToken` query parameter in the next request to retrieve the next set of results. If nextPageToken is not included in the response, all available resources have been returned.

Example: CiQxNjc0MjhIYi03OTM2LTRlYtItYTFmYi0yM2IxYWM2ZTk0NTQSAA==

isCanAlterReflections Boolean

If the current user has project-level privileges to alter Reflections, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of Objects in the `data` Array

createdAt String

Date and time that the Reflection was created, in UTC format.

Example: 2022-07-05T19:19:40.244Z

updatedAt String

Date and time that the Reflection was last updated, in UTC format.

Example: 2023-01-13T19:46:01.313Z

id String (UUID)

Unique identifier of the Reflection.

Example: 27077c03-ae49-454c-a7bb-a9a8b5eca224

reflectionType String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio UI, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: NYC_taxi_agg

currentSizeBytes Integer

Data size of the latest Reflection job (if one exists), in bytes.

Example: 9272

outputRecords Integer

Number of records returned for the latest Reflection.

Example: 51

totalSizeBytes Integer

Data size of all Reflection jobs that have not been pruned (if any exist), in bytes.

Example: 9272

datasetId String

Unique identifier of the anchor dataset that is associated with the Reflection.

Example: fa7c487f-9550-474e-8a41-4826564c6b09

datasetType String

Type for the anchor dataset that is associated with the Reflection. If the anchor dataset is a table, the type is `PHYSICAL_DATASET`. If the anchor dataset is a view, the type is `VIRTUAL_DATASET`.

Enum: PHYSICAL_DATASET, VIRTUAL_DATASET

Example: VIRTUAL_DATASET

datasetPath String

Path to the anchor dataset that is associated with the Reflection within Dremio, expressed in an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the dataset itself as the last item in the array.

Example: ["Samples","samples.dremio.com","NYC Taxi Trips"]

status Object

Information about the status of the Reflection.

Example: {
 "configStatus": "OK",
 "refreshStatus": "MANUAL",
 "availabilityStatus": "AVAILABLE",
 "combinedStatus": "CAN_ACCELERATE",
 "refreshMethod": "FULL",
 "failureCount": 0,
 "lastDataFetchAt": "2023-01-13T19:05:03.532Z",
 "expiresAt": "3022-05-16T19:46:02.342Z",
 "lastRefreshDurationMillis": 46387
}

consideredCount Integer

Number of jobs that considered the Reflection during planning.

Example: 202

matchedCount Integer

Number of jobs that matched the Reflection during planning.

Example: 45

chosenCount Integer

Number of jobs accelerated by the Reflection.

Example: 5

consideredJobsLink String

Link to list of considered jobs for the Reflection.

Example:

/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D

matchedJobsLink String

Link to list of matched jobs for the Reflection.

Example:

/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D

chosenJobsLink String

Link to list of chosen jobs for the Reflection.

Example:

/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%

22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATI
ON%22%5D%7D

isArrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, the value is ``true``. Otherwise, the value is ``false``.

Example: false

isCanView Boolean

If you can view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

isCanAlter Boolean

If you can create, edit, and view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

isEnabled Boolean

If the Reflection is available for accelerating queries, the value is ``true``. Otherwise, the value is ``false``.

Example: true

Attributes of the ``status`` Object

configStatus String

Status of the Reflection configuration. If the value is ``OK``, the Reflection configuration is free of errors. If the value is ``INVALID``, the Reflection configuration contains one or more errors.

Enum: OK, INVALID

Example: OK

refreshStatus String

Status of the Reflection refresh.

``GIVEN_UP``: Dremio attempted to refresh the Reflection multiple times, but each attempt has failed and Dremio will not make further attempts.

``MANUAL``: Refresh period is set to 0, so you must use the Dremio UI to manually refresh the Reflection.

``RUNNING``: Dremio is currently refreshing the Reflection.

``SCHEDULED``: The Reflection refreshes according to a schedule.

``ON_DATA_CHANGES``: All of the Reflection's underlying tables are in Iceberg format, and the Reflection refreshes automatically if new snapshots are created after an update to the underlying tables.

Enum: GIVEN_UP, MANUAL, RUNNING, SCHEDULED, ON_DATA_CHANGES

Example: MANUAL

availabilityStatus String

Status of the Reflection's availability for accelerating queries.

Enum: NONE, INCOMPLETE, EXPIRED, AVAILABLE

Example: AVAILABLE

combinedStatus String

Status of the Reflection based on a combination of config, refresh, and availability.

``CAN_ACCELERATE``: The Reflection is fully functional.

``CAN_ACCELERATE_WITH_FAILURES``: The most recent refresh failed to obtain a status, but Dremio still has a valid materialization.

``CANNOT_ACCELERATE_INITIALIZING``: The Reflection is currently being loaded into the materialization cache. During this time, the Reflection is unable to accelerate queries.

``CANNOT_ACCELERATE_MANUAL``: The Reflection is unable to accelerate any queries, and the ``Never Refresh`` option is selected for the refresh policy.

``CANNOT_ACCELERATE_SCHEDULED``: The Reflection is currently unable to accelerate any queries, but it has been scheduled for a refresh at a future time.

``DISABLED``: The Reflection has been manually disabled.

``EXPIRED``: The Reflection has expired and cannot be used.

``FAILED``: The attempt to refresh the Reflection has failed, typically three times in a row. The Reflection is still usable.

``INVALID``: The Reflection is invalid because the underlying dataset has changed.

``INCOMPLETE``: One or more pseudo-distributed file system (PDFS) nodes that contain materialized files are down (PFDS is supported for v21 and earlier). Only partial data is available. Configurations that use the Hadoop Distributed File System (HDFS) to store Reflections should not experience incomplete status.

``REFRESHING``: The Reflection is currently being refreshed.

Example: CAN_ACCELERATE

refreshMethod String

The method used for the most recent refresh of the Reflection. For new Reflections, the value is ``NONE`` until planned. For more information, read [Refreshing Reflections](#).

Enum: NONE, FULL, INCREMENTAL

Example: FULL

failureCount Integer

Number of times that an attempt to refresh the Reflection failed.

Example: 0

lastFailureMessage String

The error message from the last failed Reflection refresh. If the refresh of a Reflection never fails or succeeds after a failure, this attribute does not appear.

Example: "The Default engine is not online."

lastDataFetchAt String

Date and time that the Reflection data was last refreshed, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetchAt value is `1969-12-31T23:59:59.999Z`.

Example: 2023-01-13T19:05:03.532Z

expiresAt String

Date and time that the Reflection expires, in UTC format. If the Reflection is running, failing, or disabled, the expiresAt value is `1969-12-31T23:59:59.999Z`.

Example: 3022-05-16T19:46:02.342Z

lastRefreshDurationMillis Integer

Duration of the most recent refresh for the Reflection. In milliseconds.

Example: 46387

Retrieving a Reflection Summary

Retrieve a summary of the raw and aggregation Reflections in the Dremio instance.

Method and URL

```
GET /api/v3/reflection-summary
```

Parameters

pageToken Query String Optional

Token for retrieving the next page of Reflection summary results. If the Dremio instance has more Reflection summary results than the maximum per page (default 50), the response includes a nextPageToken after the data array. Use the nextPageToken value in your request URL as the pageToken value. Do not change any other query parameters included in the request URL when you use pageToken. For more information, read [pageToken Query Parameter](#).

maxResults Query Integer Optional

Maximum number of Reflection summaries to return in the response. Maximum valid value is `100`. Default is `50`. For more information, read [maxResults Query Parameter](#).

filter Query Object Optional

Filters for Reflection name, dataset name, availability status, and refresh status. Value is a URL-encoded string that represents a JSON object. The JSON object specifies the attributes to filter on and the values to match for each attribute. Available filter attributes:

reflectionType: `RAW`, `AGGREGATION` (array of string)

refreshStatus: `GIVEN_UP`, `MANUAL`, `RUNNING`, `SCHEDULED`, `ON_DATA_CHANGES` (array of string)

availabilityStatus: `NONE`, `INCOMPLETE`, `EXPIRED`, `AVAILABLE` (array of string)

configStatus: `OK`, `INVALID` (array of string)

enabledFlag: `true`, `false` (Boolean)

reflectionNameOrDatasetPath: full or partial Reflection name or dataset path; case insensitive (string)

reflectionIds: IDs of Reflections to retrieve (array of string); must be used alone, with no other filters or query parameters

For more information, read [filter Query Parameter](#).

orderBy Query String Optional

Organize the response in ascending (default) or descending order by reflectionName, datasetName, or reflectionType. To specify descending order, precede the orderBy value with a `-` character. For more information, read [orderBy Query Parameter](#).

Example Request Without Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary'
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header "Content-Type: application/json"
```

Example Response

```
{
  "data": [
    {
      "createdAt": "2023-01-13T19:46:01.313Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
      "totalSizeBytes": 9272,
    }
  ]
}
```

```

"datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
"datasetType": "VIRTUAL_DATASET",
"datasetPath": [
  "Samples",
  "samples.dremio.com",
  "NYC-taxi-trips"
],
"status": {
  "configStatus": "OK",
  "refreshStatus": "MANUAL",
  "availabilityStatus": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "refreshMethod": "FULL",
  "failureCount": 0,
  "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
  "expiresAt": "3022-05-16T19:46:02.342Z",
  "lastRefreshDurationMillis": 46387
},
"consideredCount": 202,
"matchedCount": 45,
"chosenCount": 5,

"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"isArrowCachingEnabled": false,
"isCanView": true,
"isCanAlter": true,
"isEnabled": true
},
{
  "createdAt": "2023-01-26T23:27:04.281Z",
  "updatedAt": "2023-01-26T23:27:04.281Z",
  "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
  "reflectionType": "RAW",
  "name": "Raw Reflection",
  "currentSizeBytes": 0,
  "outputRecords": -1,
  "totalSizeBytes": 0,
  "datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
  "datasetType": "PHYSICAL_DATASET",
  "datasetPath": [
    "@dremio",
    "test"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "GIVEN_UP",
    "availabilityStatus": "NONE",

```

```

    "combinedStatus": "FAILED",
    "refreshMethod": "NONE",
    "failureCount": 3,
    "lastDataFetchAt": null,
    "expiresAt": null,
    "lastRefreshDurationMillis": -1
  },
  "consideredCount": 0,
  "matchedCount": 0,
  "chosenCount": 0,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:19.030Z",
  "updatedAt": "2023-01-13T19:50:19.030Z",
  "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
  "reflectionType": "AGGREGATION",
  "name": "dataset991_agg991",
  "currentSizeBytes": 9273,
  "outputRecords": 51,
  "totalSizeBytes": 9273,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
    "expiresAt": "3022-05-16T19:46:02.342Z",
    "lastRefreshDurationMillis": 11697
  },
  "consideredCount": 60,
  "matchedCount": 9,
  "chosenCount": 0,
  "consideredJobsLink":

```



```

"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:17.714Z",
    "updatedAt": "2023-01-13T19:50:17.714Z",
    "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
    "reflectionType": "RAW",
    "name": "dataset991_raw991",
    "currentSizeBytes": 818790,
    "outputRecords": 29467,
    "totalSizeBytes": 818790,
    "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
    "datasetType": "VIRTUAL_DATASET",
    "datasetPath": [
      "temp",
      "dataset991"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "MANUAL",
      "availabilityStatus": "AVAILABLE",
      "combinedStatus": "CAN_ACCELERATE",
      "refreshMethod": "FULL",
      "failureCount": 0,
      "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
      "expiresAt": "3022-05-16T19:46:02.747Z",
      "lastRefreshDurationMillis": 16666
    },
    "consideredCount": 54,
    "matchedCount": 37,
    "chosenCount": 0,
                                                                 "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,

```

```

    "isEnabled": true
  }
],
"nextPageToken": "CiQxNjc0MjhlYi030TM2LTRlYtYTFmYi0yM2IxYWM2ZTk0NTQSAA==",
"isCanAlterReflections": true
}

```

This endpoint supports query parameters that you can add to the request URL to include only specific types of Reflections in the Reflection summary, specify the maximum number of results to return, and sort the response to list Reflections in ascending or descending order.

For example, to order the Reflections within the summary in ascending order by reflectionName, add `?orderBy=reflectionName` to the request URL. For descending order, add a `-` character before the attribute name: `?orderBy=-reflectionName`.

In the same request, you can add the `filter` query parameter to retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths. The JSON object for such a filter would look like this:

Example JSON Object for Filter

```

{
  "reflectionType": ["RAW"],
  "refreshStatus": ["MANUAL", "SCHEDULED"],
  "enabledFlag": true,
  "reflectionNameOrDatasetPath": "samples.dremio.com"
}

```

However, to use the JSON object in the request URL, you must convert it to URL-encoded JSON, which looks like this:

Example JSON Object in URL-Encoded JSON

```

%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D

```

Here is an example request URL that includes both the `orderBy` and `filter` query parameters:

Example Request with orderBy and filter Query Parameters

```

curl -X GET
'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName&filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'

```

For this request, the Reflection summary in the response will include only raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths, and the Reflections will be listed in ascending order by reflectionName.

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Recommendations](/25.x/reference/api/reflections/reflection-recommendations)[Next

Role](/25.x/reference/api/roles/)

Reflection Summary Attributes

Retrieving a Reflection Summary

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/roles/privilege>

Version: 25.x

On this page

Role Privileges Enterprise

Use the Role API to retrieve information about the privileges assigned to roles.

Role Privileges Object

```
{
  "data": [
    {
      "name": "hive",
      "type": "SOURCE",
      "privilege": "SELECT"
    },
    {
      "name": "hive",
      "type": "SOURCE",
```

```

    "privilege": "CREATE_TABLE"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "INSERT"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "DROP"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "ALTER"
  },
  {
    "name": "mysql",
    "type": "SOURCE",
    "privilege": "EXTERNAL_QUERY"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "Samples.\"samples.dremio.com\"",
    "type": "FOLDER",
    "privilege": "SELECT"
  },
  {
    "name": "Samples.\"samples.dremio.com\".\"SF_incidents2016.json\"",
    "type": "DATASET",
    "privilege": "SELECT"
  },
  {
    "name": "testing",
    "type": "SPACE",
    "privilege": "SELECT"
  }
]
}

```

Role Privileges Attributes

data Array of Object

Information about the privileges the specified role has for the entities in the current organization, up to a maximum of 100 privileges. Each object in the data array describes a privilege the role has for a specific entity.

Attributes of Objects in the `data` Array

name String (UUID)

Name of the object to which the privilege applies. The name includes the objects's parent objects, if any.

Example: hive

type String

Type of the object to which the privilege applies.

Enum: SPACE, SOURCE, HOME, FOLDER, DATASET, FUNCTION

Example: SOURCE

privilege String

Name of the privilege that the role has for the object. Available privileges vary for different object types.

Example: SELECT

Retrieving Role Privileges

Retrieve the specified role's privileges.

Method and URL

```
GET /api/v3/role/{id}/privilege
```

Parameters

id Path String (UUID)

Unique identifier of the role whose privileges you want to retrieve.

Example: 3d83e7d7-98ee-4afa-ebdd-41c30eb92744

Example Request

```
curl -X GET 'https://{hostname}/api/v3/role/3d83e7d7-98ee-4afa-ebdd-41c30eb92744/privilege' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "name": "hive",
```

```

    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "CREATE_TABLE"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "INSERT"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "DROP"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "ALTER"
  },
  {
    "name": "mysql",
    "type": "SOURCE",
    "privilege": "EXTERNAL_QUERY"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "Samples.\"samples.dremio.com\"",
    "type": "FOLDER",
    "privilege": "SELECT"
  },
  {
    "name": "Samples.\"samples.dremio.com\".\"SF_incidents2016.json\"",
    "type": "DATASET",
    "privilege": "SELECT"
  },
  {
    "name": "testing",
    "type": "SPACE",
    "privilege": "SELECT"
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Role](/25.x/reference/api/roles/)[Next

Scripts](/25.x/reference/api/scripts/)

Role Privileges Attributes

Retrieving Role Privileges

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/user/privilege>

Version: 25.x

On this page

User Privileges Enterprise

Use the User API to retrieve privilege information for Dremio users.

User Privileges Object

```
{
  "data": [
    {
      "name": "\\\"@dremio\\\".\\\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\\\"",
      "type": "PDS",
      "privilege": "ALTER"
    },
    {
      "name": "\\\"@dremio\\\".\\\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\\\"",
      "type": "PDS",
      "privilege": "SELECT"
    },
    {
      "name": "\\\"@dremio\\\".Business",
      "type": "FOLDER",
      "privilege": "ALTER"
    },
    {
```

```

    "name": "\"@dremio\".Business",
    "type": "FOLDER",
    "privilege": "SELECT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "INSERT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "DROP"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "CREATE_TABLE"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "ALTER"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "SELECT"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "MANAGE_GRANTS"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "MODIFY"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "ALTER_REFLECTION"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "VIEW_REFLECTION"
  },

```



```

{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "ALTER"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "SELECT"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "ALTER"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "DELETE"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "MODIFY"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "VIEW"
}
]
}

```

User Privileges Attributes

data Array of Object

Information about the privileges the specified user has for the catalog objects in the current organization, up to a maximum of 100 privileges. Each object in the data array describes a privilege on a specific catalog object.

Attributes of Objects in the `data` Array

name String (UUID)

Name of the object to which the privilege applies. The name includes the entity's parent space or folder, if any.

Example: "@dremio"."1c0accd3-e8c0-1d55-23a2-0ff6529f6c00"

type String

Type of the object to which the privilege applies.

Enum: SPACE, SOURCE, HOME, FOLDER, PDS, VDS, FUNCTION

Example: PDS

privilege String

Name of the privilege that the user has for the object. Available privileges vary for different object types.

Example: ALTER

Retrieving User Privileges

Retrieve the specified user's privileges.

Method and URL

```
GET /api/v3/user/{id}/privilege
```

Parameters

id Path String

Unique identifier of the user whose privileges you want to retrieve.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example Request

```
curl -X GET 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/privilege' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "name": "\"@dremio\".\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\"",
```

```

    "type": "PDS",
    "privilege": "ALTER"
  },
  {
    "name": "\"@dremio\".\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\"",
    "type": "PDS",
    "privilege": "SELECT"
  },
  {
    "name": "\"@dremio\".Business",
    "type": "FOLDER",
    "privilege": "ALTER"
  },
  {
    "name": "\"@dremio\".Business",
    "type": "FOLDER",
    "privilege": "SELECT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "INSERT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "DROP"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "CREATE_TABLE"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "ALTER"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "SELECT"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "MANAGE_GRANTS"
  },
  {

```

```

    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "MODIFY"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "ALTER_REFLECTION"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "VIEW_REFLECTION"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "ALTER"
  },
  {
    "name": "\"testing\".\"NYC-taxi-trips\"",
    "type": "VDS",
    "privilege": "SELECT"
  },
  {
    "name": "\"testing\".\"NYC-taxi-trips\"",
    "type": "VDS",
    "privilege": "MANAGE_GRANTS"
  },
  {
    "name": "\"testing\".\"NYC-taxi-trips\"",
    "type": "VDS",
    "privilege": "ALTER"
  },
  {
    "name": "taxi",
    "type": "SCRIPT",
    "privilege": "MANAGE_GRANTS"
  },
  {
    "name": "taxi",
    "type": "SCRIPT",
    "privilege": "DELETE"
  },
  {
    "name": "taxi",
    "type": "SCRIPT",
    "privilege": "MODIFY"
  },
  {
    "name": "taxi",
    "type": "SCRIPT",
    "privilege": "VIEW"
  }
}

```

```
]
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

User]([25.x/reference/api/user/])[Next

User Tokens]([25.x/reference/api/user/token)

User Privileges Attributes

Retrieving User Privileges

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/user/token>

Version: 25.x

On this page

User Tokens Enterprise

Use the User API to create and retrieve personal access tokens for the current Dremio user and delete personal access tokens for any Dremio user.

note

You must [enable the use of personal access tokens](#) to make API requests for user tokens.

User Tokens Object

```
{
  "data": [
    {
      "tid": "98ec8f42-7764-4d9d-af5a-693f1f1cc444",
      "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
```

```

    "label": "Tableau",
    "createdAt": "2023-02-19T15:41:15.323Z",
    "expiresAt": "2023-03-21T15:41:15.323Z"
  },
  {
    "tid": "3b76a1e4-6539-46de-8f06-b7c41c71b61e",
    "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
    "label": "Test Nessie Source",
    "createdAt": "2023-03-02T19:39:52.159Z",
    "expiresAt": "2023-04-01T19:39:52.159Z"
  },
  {
    "tid": "9376ef58-7b4c-2419-b1cb-a4ce4c53dfa7",
    "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
    "label": "Feature Testing",
    "createdAt": "2023-03-07T14:47:08.211Z",
    "expiresAt": "2023-09-03T14:47:08.211Z"
  }
]
}

```

User Tokens Attributes

data Array of Object

Information about the user's tokens. Each object in the data array describes a different token of the user.

Attributes of Objects in the `data` Array

tid String (UUID)

Unique identifier of the token.

Example: 98ec8f42-7764-4d9d-af5a-693f1f1cc444

uid String (UUID)

Unique identifier of the user.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

label String

User-provided name of the token.

Example: Tableau

createdAt String

Date and time that the token was created, in UTC format.

Example: 2023-02-19T15:41:15.323Z

expiresAt String

Date and time that the token will expire, in UTC format.

Example: 2023-03-21T15:41:15.323Z

Creating a Token for a User

Create a personal access token for the current user.

note

You can create personal access tokens only for your own user account, and only you may use the tokens you create. Administrators cannot create tokens for other users or distribute tokens to other users.

Method and URL

```
POST /api/v3/user/{id}/token
```

Parameters

id Path String (UUID)

Unique identifier of the user.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

label Body String

User-provided name for the token

Example: Feature Testing

millisecondsToExpire Body String

Number of milliseconds until the token should expire. Maximum value is `1555200000`, which is equivalent to 180 days. If you omit the millisecondsToExpire parameter, the new token's expiresAt setting will default to the time that the token is created and the token will immediately expire.

Example: 1555200000

Example Request

```
curl -X POST 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "label": "Feature Testing",
  "millisecondsToExpire": 1555200000
}'
```

The response contains the personal access token:

Example Response

```
EXAMPLETOKEN7TjB3mfPS6AZQ5aPcXPmJS2ofXpLL86dmpDXRbKKi52BQdthnk==
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Retrieving All Tokens for a User

Retrieve all tokens for the current user.

note

You can only retrieve personal access tokens for your own user account. Administrators cannot retrieve other users' tokens.

Method and URL

```
GET /api/v3/user/{id}/token
```

Parameters

id Path String (UUID)

Unique identifier of the current user. You can only retrieve personal access tokens for your own user account.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example Request

```
curl -X GET 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
```



```

    "tid": "98ec8f42-7764-4d9d-af5a-693f1f1cc444",
    "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
    "label": "Tableau",
    "createdAt": "2023-02-19T15:41:15.323Z",
    "expiresAt": "2023-03-21T15:41:15.323Z"
  },
  {
    "tid": "3b76a1e4-6539-46de-8f06-b7c41c71b61e",
    "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
    "label": "Test Nessie Source",
    "createdAt": "2023-03-02T19:39:52.159Z",
    "expiresAt": "2023-04-01T19:39:52.159Z"
  },
  {
    "tid": "9376ef58-7b4c-2419-b1cb-a4ce4c53dfa7",
    "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
    "label": "Feature Testing",
    "createdAt": "2023-03-07T14:47:08.211Z",
    "expiresAt": "2023-09-03T14:47:08.211Z"
  }
]
}

```

note

If the user has no personal access tokens, the response contains an empty data array.

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Deleting All Tokens for a User

Delete all tokens for the specified user.

note

You must be a member of the ADMIN role to delete other users' tokens.

Method and URL

```
DELETE /api/v3/user/{id}/token
```

Parameters

id Path String (UUID)

Unique identifier of the user whose tokens you want to delete.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Deleting a Token for a User

Delete the specified token for the specified user.

note

You must be a member of the ADMIN role to delete other users's tokens.

Method and URL

```
DELETE /api/v3/user/{id}/token/{token-id}
```

Parameters

id Path String (UUID)

Unique identifier of the user whose token you want to delete.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

token-id Path String (UUID)

Unique identifier of the token you want to delete.

Example: 98ec8f42-7764-4d9d-af5a-693f1f1cc444

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token/98ec8f42-7764-4d9d-af5a-693f1f1cc444' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

User Privileges](/25.x/reference/api/user/privilege)[Next

Workload Management](/25.x/reference/api/wlm/)

User Tokens Attributes

Creating a Token for a User

Parameters

Response Status Codes

Retrieving All Tokens for a User

Parameters

Response Status Codes

Deleting All Tokens for a User

Parameters

Response Status Codes

Deleting a Token for a User

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/wlm/queue>

Version: 25.x

On this page

Queue Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM queues.

Queue Object

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",
      "name": "High Cost Reflections",
      "maxMemoryPerNodeBytes": 8589934592,
      "maxQueryMemoryPerNodeBytes": 8589934592,
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 300000,
      "maxRunTimeoutMs": 300000,
      "engineId": "DATA"
    },
    {
      "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "tag": "HM2D9XElG3U=",
      "name": "Low Cost Reflections",
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 86400000
    },
    {
      "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "tag": "p22KaFcaB7g=",
      "name": "COPY & OPTIMIZATION Queue",
      "maxMemoryPerNodeBytes": 4294967296,
      "maxQueryMemoryPerNodeBytes": 4294967296,
      "cpuTier": "MEDIUM",
      "maxAllowedRunningJobs": 2,
      "maxStartTimeoutMs": 300000,
      "engineId": "YARN"
    },
    {
      "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "tag": "//gNL3Ta2bY=",
      "name": "Low Cost User Queries",
      "cpuTier": "MEDIUM",
      "maxAllowedRunningJobs": 100,
    }
  ]
}
```

```

    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Queue Attributes

data Array of Object

List of queue objects in the Dremio instance.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the queue, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag String

Unique identifier of the version of the queue. Dremio changes the tag whenever the queue changes and uses the tag to ensure that PUT requests apply to the most recent version of the queue.

Example: BNGRmgfEnDg=

name String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Integer

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

maxQueryMemoryPerNodeBytes Integer

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier String

Amount of CPU time that threads get compared to other threads.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Integer

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Integer

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Integer

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId String

Name of the execution engine to which the queue's queries are routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. The engineId attribute is omitted from the queue object if no engine is specified. For more information, read [Workload Management-based Routing](#).

Example: DATA

Creating a Queue

Create a WLM queue.

Method and URL

```
POST /api/v3/wlm/queue
```

Parameters

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId Body String Optional

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. For more information, read [Workload Management-based Routing](#).

Example: DATA

Example Request

```
curl -X POST 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "High Cost Reflections",
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "maxMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

409 Conflict

Retrieving All Queues

Retrieve all WLM queues.

Method and URL

```
GET /api/v3/wlm/queue
```

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",

```



```

    "name": "High Cost Reflections",
    "maxMemoryPerNodeBytes": 8589934592,
    "maxQueryMemoryPerNodeBytes": 8589934592,
    "cpuTier": "BACKGROUND",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000,
    "maxRunTimeoutMs": 300000,
    "engineId": "DATA"
  },
  {
    "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "tag": "HM2D9XElG3U=",
    "name": "Low Cost Reflections",
    "cpuTier": "BACKGROUND",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 86400000
  },
  {
    "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "tag": "p22KaFcaB7g=",
    "name": "COPY & OPTIMIZATION Queue",
    "maxMemoryPerNodeBytes": 4294967296,
    "maxQueryMemoryPerNodeBytes": 4294967296,
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 2,
    "maxStartTimeoutMs": 300000,
    "engineId": "YARN"
  },
  {
    "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "tag": "//gNL3Ta2bY=",
    "name": "Low Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 100,
    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a Queue by ID

Retrieve a specific WLM queue by the queue's ID.

Method and URL

```
GET /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to retrieve, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a Queue by Name

Retrieve a specific WLM queue by the queue's name.

Method and URL

```
GET /api/v3/wlm/queue/by-name/{name}
```

Parameters

name Path String

Name for the queue you want to retrieve. If the queue name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: High%20Cost%20Reflections

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/by-name/High%20Cost%20Reflections' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized
403 Forbidden
404 Not Found
500 Internal Server Error

Updating a Queue

Update the specified WLM queue.

Method and URL

```
PUT /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to update, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag Body String

Unique identifier of the version of the queue to update. Dremio uses the tag to ensure that you are updating the most recent version of the queue.

Example: BNGRmgfEnDg=

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node. If you omit the maxMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node. If you omit the maxQueryMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxQueryMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxQueryMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`. If you omit the cpuTier parameter in a PUT request, Dremio replaces it with the default value. To keep the existing setting while making other updates, include the existing cpuTier parameter and setting in the PUT request.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: LOW

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel. If you omit the maxAllowedRunningJobs parameter in a PUT request, Dremio removes the existing maxAllowedRunningJobs value from the queue. To keep the existing value while making other updates, include the existing maxAllowedRunningJobs parameter and value in the PUT request.

Example: 100

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds. If you omit the maxStartTimeoutMs parameter in a PUT request, Dremio removes the existing maxStartTimeoutMs setting from the queue. To keep the existing setting while making other updates, include the existing maxStartTimeoutMs parameter and setting in the PUT request.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query is allowed to run before it is cancelled, in milliseconds. If you omit the maxRunTimeoutMs parameter in a PUT request, Dremio removes the existing maxRunTimeoutMs value from the queue. To keep the existing value while making other updates, include the existing maxRunTimeoutMs parameter and value in the PUT request.

Example: 300000

engineId Body String

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. For more information, read [Workload Management-based Routing](#).

Example: DATA

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
```

```
"tag": "BNGRmgfEnDg=",
"name": "High Cost Reflections",
"maxMemoryPerNodeBytes": 8589934592,
"maxQueryMemoryPerNodeBytes": 8589934592,
"cpuTier": "LOW",
"maxAllowedRunningJobs": 100,
"maxStartTimeoutMs": 300000,
"maxRunTimeoutMs": 300000,
"engineId": "DATA"
}'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "xQh6KNyEjus=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "LOW",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

500 Internal Server Error

Deleting a Queue

Delete the specified WLM queue.

Method and URL

```
DELETE /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue that you want to delete, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Was this page helpful?

[\[Previous](#)

[Workload Management\]\(/25.x/reference/api/wlm/\)](#)[\[Next](#)

[Rule\]\(/25.x/reference/api/wlm/rule\)](#)

[Queue Attributes](#)

[Creating a Queue](#)

[Parameters](#)

[Response Status Codes](#)

[Retrieving All Queues](#)

[Response Status Codes](#)

[Retrieving a Queue by ID](#)

Parameters

Response Status Codes

Retrieving a Queue by Name

Parameters

Response Status Codes

Updating a Queue

Parameters

Response Status Codes

Deleting a Queue

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/wlm/rule>

Version: 25.x

On this page

Rule Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM rules.

The rule object includes a rules array (also called the ruleset). Each object in the rules array represents an individual rule. Dremio processes rules in the order they are listed within the rules array: the highest-priority rule is listed first, and the lowest-priority rule is listed last.

Rule Object

```
{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
```



```

    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Rule Attributes

tag String

Unique identifier of the version of the rule. Dremio changes the tag whenever a rule changes and uses the tag to ensure that PUT requests apply to the most recent version of the rules.

Example: VmqwaZ90VY4=

rules Array of Object

List of rule objects in the Dremio instance.

defaultRule Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule.

Example: `{"name": "All Other Queries", "action": "REJECT", "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}`

Attributes of Objects in the `rules` Array

name String

User-provided name for the rule.

Example: High Cost Reflections

conditions String

Conditions that queries must match to be placed in the queue.

Example: `query_type() = 'Reflections' AND query_cost() >= 30000000`

acceptId String (UUID)

Unique identifier of the queue in which the rule places queries.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

acceptName Integer

User-provided name for the queue in which the rule places queries.

Example: High Cost Reflections

rejectMessage Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Integer

Action the rule takes for queries that match the rule conditions.

Enum: PLACE, REJECT

Example: PLACE

id String (UUID)

Unique identifier of the rule, in UTC format.

Example: fa1ec87d-923b-414c-9064-e079f39f5c49

Attributes of the `defaultRule` Object

name String

User-provided name for the default rule.

Example: All Other Queries

acceptId String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action String

Action the default rule takes for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

Creating or Updating a Rule

Create or update a WLM rule.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To add or update an individual rule, you must include the entire rules array in the request body. It is not necessary to specify the ID for the rule you want to delete in the request URL.

Method and URL

```
PUT /api/v3/wlm/rule
```

Parameters

tag Body String

Unique identifier of the rules instance. Dremio uses the tag to ensure that PUT requests apply to the most recent version of the rules. Omit if you are creating rules for the organization for the first time.

Example: VmqwaZ90VY4=

rules Body Array of Object

List of rule objects in the Dremio instance. To add or update an individual rule, you must include the entire rules array in the request body.

Example:

defaultRule Body Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule. To add or update an individual rule, you must include the defaultRule object in the request body.

Example: {"name": "All Other Queries", "action": "REJECT", "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}

Parameters of Objects in the `rules` Array

name Body String

User-provided name for the rule.

Example: DevOps and Engineering

conditions Body String

Conditions that queries must match to be placed in the queue.

Example: is_member('DevOps') OR is_member('Engineering')

acceptId Body String (UUID)

For rules whose action is `PLACE`, the unique identifier of the queue in which the rule should place queries.

Example: b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1

acceptName Body Integer Optional

For rules whose action is `PLACE`, the user-provided name for the queue in which the rule should place queries.

Example: High Cost Reflections

rejectMessage Body Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Body Integer Optional

Action the rule should take for queries that match the rule conditions. Default is `PLACE`.

Enum: PLACE, REJECT

Example: PLACE

Parameters of the `defaultRule` Object

name Body String

User-provided name for the default rule.

Example: All Other Queries

acceptId Body String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Body Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Body Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action Body String

Action the default rule should take for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id Body String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

This example request demonstrates how to add a `DevOps and Engineering` rule to the ruleset:

Example Request to Add a Rule

```

curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ]
},

```

```
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'
```

Example Response

```
{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE",
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {

```

```

    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

To change the order in which Dremio processes rules, send a PUT request that lists the rules in the desired order within the rules array, with the highest-priority rule listed first, and the lowest-priority rule listed last.

This example request reorders the rules so that the `COPY & OPTIMIZATION Rule` will be the highest-priority rule:

Example Request to Reorder Rules

```

curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqwaZ90VY4=",
  "rules": [
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    }
  ],
}

```



```
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 300000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'
```

Example Response

```
{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    }
  ]
}
```

```
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 300000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieving All Rules

Retrieve all WLM rules.

Method and URL

```
GET /api/v3/wlm/rule
```

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE",
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ],
  "defaultRule": {
```

```
"name": "All Other Queries",  
"action": "REJECT",  
"id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"  
}  
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Deleting a Rule

Delete a WLM rule from the ruleset.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To delete a rule, send a PUT request that omits the rule from the rules array. It is not necessary to specify the ID for the rule you want to delete in the request URL.

note

The default rule can be updated but cannot be deleted.

Method and URL

```
PUT /api/v3/wlm/rule/
```

This example request demonstrates how to remove the `DevOps and Engineering` rule added in the Creating or Updating a Rule example:

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json' \  
--data-raw '{  
  "tag": "3uzixTFD134=",  
  "rules": [  
    {  
      "name": "High Cost Reflections",  
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",  
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",  
      "acceptName": "High Cost Reflections",  
      "action": "PLACE",
```

```

    "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
  },
  {
    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'

```

Example Response

```

{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",

```

```

    "id": "falec87d-923b-414c-9064-e079f39f5c49"
  },
  {
    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Was this page helpful?

[Previous

Queue](/25.x/reference/api/wlm/queue)[Next

SQL Reference](/25.x/reference/sql/)

Rule Attributes

Creating or Updating a Rule

Parameters

Response Status Codes

Retrieving All Rules

Response Status Codes

Deleting a Rule

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/wlm/queue/>

Version: 25.x

On this page

Queue Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM queues.

Queue Object

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",
      "name": "High Cost Reflections",
      "maxMemoryPerNodeBytes": 8589934592,
      "maxQueryMemoryPerNodeBytes": 8589934592,
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 300000,
      "maxRunTimeoutMs": 300000,
      "engineId": "DATA"
    },
    {
```

```

    "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "tag": "HM2D9XElG3U=",
    "name": "Low Cost Reflections",
    "cpuTier": "BACKGROUND",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 86400000
  },
  {
    "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "tag": "p22KaFcaB7g=",
    "name": "COPY & OPTIMIZATION Queue",
    "maxMemoryPerNodeBytes": 4294967296,
    "maxQueryMemoryPerNodeBytes": 4294967296,
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 2,
    "maxStartTimeoutMs": 300000,
    "engineId": "YARN"
  },
  {
    "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "tag": "//gNL3Ta2bY=",
    "name": "Low Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 100,
    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Queue Attributes

data Array of Object

List of queue objects in the Dremio instance.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the queue, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag String

Unique identifier of the version of the queue. Dremio changes the tag whenever the queue changes and uses the tag to ensure that PUT requests apply to the most recent version of the queue.

Example: BNGRmgfEnDg=

name String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Integer

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

maxQueryMemoryPerNodeBytes Integer

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier String

Amount of CPU time that threads get compared to other threads.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Integer

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Integer

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Integer

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId String

Name of the execution engine to which the queue's queries are routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. The engineId attribute is omitted from the queue object if no engine is specified. For more information, read [Workload Management-based Routing](#).

Example: DATA

Creating a Queue

Create a WLM queue.

Method and URL

```
POST /api/v3/wlm/queue
```

Parameters

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId Body String Optional

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. For more information, read [Workload Management-based Routing](#).

Example: DATA

Example Request

```
curl -X POST 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "High Cost Reflections",
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "maxMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Retrieving All Queues

Retrieve all WLM queues.

Method and URL

```
GET /api/v3/wlm/queue
```

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",
      "name": "High Cost Reflections",
      "maxMemoryPerNodeBytes": 8589934592,
      "maxQueryMemoryPerNodeBytes": 8589934592,
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 300000,
      "maxRunTimeoutMs": 300000,
      "engineId": "DATA"
    },
    {
      "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "tag": "HM2D9XElG3U=",
      "name": "Low Cost Reflections",
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 86400000
    },
    {
      "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "tag": "p22KaFcaB7g=",
      "name": "COPY & OPTIMIZATION Queue",
      "maxMemoryPerNodeBytes": 4294967296,
      "maxQueryMemoryPerNodeBytes": 4294967296,
      "cpuTier": "MEDIUM",
      "maxAllowedRunningJobs": 2,
      "maxStartTimeoutMs": 300000,
      "engineId": "YARN"
    },
    {
```

```

    "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "tag": "//gNL3Ta2bY=",
    "name": "Low Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 100,
    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a Queue by ID

Retrieve a specific WLM queue by the queue's ID.

Method and URL

```
GET /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to retrieve, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example Request

```

curl -X GET 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <PersonalAccessToken>' \

```

```
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieving a Queue by Name

Retrieve a specific WLM queue by the queue's name.

Method and URL

```
GET /api/v3/wlm/queue/by-name/{name}
```

Parameters

name Path String

Name for the queue you want to retrieve. If the queue name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: High%20Cost%20Reflections

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/by-name/High%20Cost%20Reflections' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Updating a Queue

Update the specified WLM queue.

Method and URL

```
PUT /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to update, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag Body String

Unique identifier of the version of the queue to update. Dremio uses the tag to ensure

that you are updating the most recent version of the queue.

Example: BNGRmgfEnDg=

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node. If you omit the maxMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node. If you omit the maxQueryMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxQueryMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxQueryMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`. If you omit the cpuTier parameter in a PUT request, Dremio replaces it with the default value. To keep the existing setting while making other updates, include the existing cpuTier parameter and setting in the PUT request.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: LOW

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel. If you omit the maxAllowedRunningJobs parameter in a PUT request, Dremio removes the existing maxAllowedRunningJobs value from the queue. To keep the existing value while making other updates, include the existing maxAllowedRunningJobs parameter and value in the PUT request.

Example: 100

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds. If you omit the maxStartTimeoutMs parameter in a PUT request, Dremio removes the existing maxStartTimeoutMs setting from the queue. To keep the existing setting while making other updates, include the existing maxStartTimeoutMs parameter and setting in the PUT request.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query is allowed to run before it is cancelled, in milliseconds. If you omit the maxRunTimeoutMs parameter in a PUT request, Dremio removes the existing maxRunTimeoutMs value from the queue. To keep the existing value while making other updates, include the existing maxRunTimeoutMs parameter and value in the PUT request.

Example: 300000

engineId Body String

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. For more information, read [Workload Management-based Routing](#).

Example: DATA

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "LOW",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Example Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "xQh6KNyEjus=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "LOW",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

500 Internal Server Error

Deleting a Queue

Delete the specified WLM queue.

Method and URL

```
DELETE /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue that you want to delete, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example Request

```
curl -X DELETE \
'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response

No response

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Workload Management](/25.x/reference/api/wlm/)[Next

Rule](/25.x/reference/api/wlm/rule)

Queue Attributes

Creating a Queue

Parameters

Response Status Codes

Retrieving All Queues

Response Status Codes

Retrieving a Queue by ID

Parameters

Response Status Codes

Retrieving a Queue by Name

Parameters

Response Status Codes

Updating a Queue

Parameters

Response Status Codes

Deleting a Queue

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/25.x/reference/api/wlm/rule/>

Version: 25.x

On this page

Rule Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM rules.

The rule object includes a rules array (also called the ruleset). Each object in the rules array represents an individual rule. Dremio processes rules in the order they are listed within the rules array: the highest-priority rule is listed first, and the lowest-priority rule is listed last.

Rule Object

```
{
  "tag": "VmqwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY', 'OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 300000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 300000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ]
}
```

```

    },
    "defaultRule": {
      "name": "All Other Queries",
      "action": "REJECT",
      "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
    }
  }
}

```

Rule Attributes

tag String

Unique identifier of the version of the rule. Dremio changes the tag whenever a rule changes and uses the tag to ensure that PUT requests apply to the most recent version of the rules.

Example: VmqwaZ90VY4=

rules Array of Object

List of rule objects in the Dremio instance.

defaultRule Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule.

Example: {"name": "All Other Queries", "action": "REJECT", "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}

Attributes of Objects in the `rules` Array

name String

User-provided name for the rule.

Example: High Cost Reflections

conditions String

Conditions that queries must match to be placed in the queue.

Example: query_type() = 'Reflections' AND query_cost() >= 30000000

acceptId String (UUID)

Unique identifier of the queue in which the rule places queries.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

acceptName Integer

User-provided name for the queue in which the rule places queries.

Example: High Cost Reflections

rejectMessage Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Integer

Action the rule takes for queries that match the rule conditions.

Enum: PLACE, REJECT

Example: PLACE

id String (UUID)

Unique identifier of the rule, in UTC format.

Example: fa1ec87d-923b-414c-9064-e079f39f5c49

Attributes of the `defaultRule` Object

name String

User-provided name for the default rule.

Example: All Other Queries

acceptId String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action String

Action the default rule takes for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

Creating or Updating a Rule

Create or update a WLM rule.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To add or update an individual rule, you must include the entire rules array in the request body. It is not necessary to specify the ID for the rule you want to delete in the request URL.

Method and URL

```
PUT /api/v3/wlm/rule
```

Parameters

tag Body String

Unique identifier of the rules instance. Dremio uses the tag to ensure that PUT requests apply to the most recent version of the rules. Omit if you are creating rules for the organization for the first time.

Example: VmqwaZ90VY4=

rules Body Array of Object

List of rule objects in the Dremio instance. To add or update an individual rule, you must include the entire rules array in the request body.

Example:

defaultRule Body Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule. To add or update an individual rule, you must include the defaultRule object in the request body.

Example: {"name": "All Other Queries", "action": "REJECT", "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}

Parameters of Objects in the `rules` Array

name Body String

User-provided name for the rule.

Example: DevOps and Engineering

conditions Body String

Conditions that queries must match to be placed in the queue.

Example: is_member('DevOps') OR is_member('Engineering')

acceptId Body String (UUID)

For rules whose action is `PLACE`, the unique identifier of the queue in which the rule should place queries.

Example: b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1

acceptName Body Integer Optional

For rules whose action is `PLACE`, the user-provided name for the queue in which the rule should place queries.

Example: High Cost Reflections

rejectMessage Body Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Body Integer Optional

Action the rule should take for queries that match the rule conditions. Default is `PLACE`.

Enum: PLACE, REJECT

Example: PLACE

Parameters of the `defaultRule` Object

name Body String

User-provided name for the default rule.

Example: All Other Queries

acceptId Body String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Body Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Body Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action Body String

Action the default rule should take for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id Body String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

This example request demonstrates how to add a `DevOps and Engineering` rule to the ruleset:

Example Request to Add a Rule

```
curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    }
  ]
}
```

```

    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8elf-4ee2e8438f81"
}
}'

```

Example Response

```

{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE",
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
    },
  ],
  {

```

```

    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 30000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 30000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

To change the order in which Dremio processes rules, send a PUT request that lists the rules in the desired order within the rules array, with the highest-priority rule listed first, and the lowest-priority rule listed last.

This example request reorders the rules so that the `COPY & OPTIMIZATION Rule` will be the highest-priority rule:

Example Request to Reorder Rules

```

curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqlwaZ90VY4=",
  "rules": [

```

```

{
  "name": "COPY & OPTIMIZATION Rule",
  "conditions": "query_label() in ('COPY','OPTIMIZATION')",
  "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
  "acceptName": "COPY & OPTIMIZATION Queue",
  "action": "PLACE",
  "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
},
{
  "name": "High Cost Reflections",
  "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
  "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "acceptName": "High Cost Reflections",
  "action": "PLACE",
  "id": "falec87d-923b-414c-9064-e079f39f5c49"
},
{
  "name": "Low Cost Reflections",
  "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
  "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
  "acceptName": "Low Cost Reflections",
  "action": "PLACE",
  "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
},
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 30000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 30000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8elf-4ee2e8438f81"
}
}'

```

Example Response

```

{
  "tag": "3uzixTFD134=",
  "rules": [

```

```

{
  "name": "COPY & OPTIMIZATION Rule",
  "conditions": "query_label() in ('COPY','OPTIMIZATION')",
  "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
  "acceptName": "COPY & OPTIMIZATION Queue",
  "action": "PLACE",
  "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
},
{
  "name": "High Cost Reflections",
  "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
  "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "acceptName": "High Cost Reflections",
  "action": "PLACE",
  "id": "falec87d-923b-414c-9064-e079f39f5c49"
},
{
  "name": "Low Cost Reflections",
  "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
  "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
  "acceptName": "Low Cost Reflections",
  "action": "PLACE",
  "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
},
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 30000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 30000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8elf-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
409 Conflict

Retrieving All Rules

Retrieve all WLM rules.

Method and URL

```
GET /api/v3/wlm/rule
```

Example Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/rule' \  
--header 'Authorization: Bearer <PersonalAccessToken>' \  
--header 'Content-Type: application/json'
```

Example Response

```
{  
  "tag": "3uzixTFD134=",  
  "rules": [  
    {  
      "name": "DevOps and Engineering",  
      "conditions": "is_member('DevOps') OR is_member('Engineering')",  
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",  
      "acceptName": "DevOps and Eng Testing",  
      "action": "PLACE",  
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"  
    },  
    {  
      "name": "High Cost Reflections",  
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",  
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",  
      "acceptName": "High Cost Reflections",  
      "action": "PLACE",  
      "id": "falec87d-923b-414c-9064-e079f39f5c49"  
    },  
    {  
      "name": "Low Cost Reflections",  
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",  
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",  
      "acceptName": "Low Cost Reflections",  
      "action": "PLACE",  
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"  
    }  
  ],  
}
```

```

{
  "name": "COPY & OPTIMIZATION Rule",
  "conditions": "query_label() in ('COPY','OPTIMIZATION')",
  "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
  "acceptName": "COPY & OPTIMIZATION Queue",
  "action": "PLACE",
  "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
},
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 300000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Deleting a Rule

Delete a WLM rule from the ruleset.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To delete a rule, send a PUT request that omits the rule from the rules array. It is not necessary to specify the ID for the rule you want to delete in the request URL.

note

The default rule can be updated but cannot be deleted.

Method and URL

```
PUT /api/v3/wlm/rule/
```

This example request demonstrates how to remove the `DevOps and Engineering` rule added in the Creating or Updating a Rule example:

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
```



```

    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'

```

Example Response

```

{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 300000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 300000000",

```

```

    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Was this page helpful?

[\[Previous](#)

[Queue\]\(/25.x/reference/api/wlm/queue\)\[Next](#)

[SQL Reference\]\(/25.x/reference/sql/\)](#)

[Rule Attributes](#)

[Creating or Updating a Rule](#)

[Parameters](#)

[Response Status Codes](#)

[Retrieving All Rules](#)

[Response Status Codes](#)

[Deleting a Rule](#)

[Response Status Codes](#)

Source: dremio-software-v26-api.md

Dremio API Documentation
dremio-software-v26-api.md

-

Source:
<https://docs.dremio.com/current/reference/api/>

Version: current [26.x]

On this page

API Reference

The Dremio REST API is organized by resource types such as ``sources`` and is designed around RESTful principles.

``GET`` is used to retrieve existing resources

``POST`` creates new resources

``PUT`` updates resources

``DELETE`` removes resources

Base URL

All API URLs referenced in this documentation have the following base URL unless otherwise specified:

Base URL

```
{hostname}/api/v3
```

Versions prior to ``v3`` are considered internal and are subject to change without notice.

In this documentation, curly braces (``{}``) are used to indicate sections of URLs where you have to supply a value. For example:

User-supplied values in URLs

```
/api/v3/source/{id}
```

Authentication

Each REST request requires an authorization header with a Dremio access token to authenticate the requester unless otherwise indicated. Dremio accepts three types of access tokens for authenticating REST requests.

OAuth access tokens are created in Dremio using the Dremio REST API.

Personal access tokens are created in the Dremio console or with REST.

Authentication tokens are generated from a username and password using the Dremio v2 API.

All Dremio access tokens are Bearer tokens and can be used in the REST authorization

header of each REST request.

Example Dremio REST request

```
curl -X GET 'https://{hostname}/api/v3/{path_to_endpoint}' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

OAuth Access Tokens Enterprise

Users can create OAuth access tokens by exchanging a local or LDAP username and password, a PAT, or an external JWT using the `/oauth/token` REST API. Dremio provides [sample code](#) for each of these cases.

Dremio recommends OAuth access tokens to improve system security:

OAuth access tokens are typically short-lived, reducing the window of opportunity for attackers if a token is compromised.

Users must manually revoke compromised or suspected PATs, often leading to forgotten, unused tokens.

Personal Access Tokens Enterprise

Any user can create personal access tokens (PATs) [in the Dremio console](#) or [using REST](#). Users can configure the lifetime of each personal access token, from 1 day to a maximum defined by the `auth.personal-access-token.max_lifetime_days` [support setting](#).

Dremio Authentication Tokens

Users can generate authentication tokens from their Dremio username and password. Authentication tokens have a nonconfigurable lifetime of 30 hours.

caution

Generating an authentication token requires API v2. API versions prior to v3 are considered internal and are subject to change without notice.

To generate an authentication token:

Send an API request to the login URL with your Dremio username and password in the request body.

Example

Request

```
curl -X POST 'http://{hostname}/apiv2/login' \
--header 'Content-Type: application/json' \
--data-raw '{
"userName": "dremio",
```

```
"password": "dremio123"
}'
```

Response

```
{
  "token": "4ksrt534vk7fkq64xh55g7776b",
  "userName": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "expires": 1686578200000,
  "email": "dremio@dremio.test",
  "userId": "5a679dd5-52d7-402a-871d-7fbee3fe8007",
  "admin": true,
  "clusterId": "7468ce46-58af-4dce-a42f-4c51048968f5",
  "clusterCreatedAt": 1681311939728,
  "version": "24.0.0-main-202305040803350903-2d5579e3",
  "permissions": {
    "canUploadProfiles": true,
    "canDownloadProfiles": true,
    "canEmailForSupport": true,
    "canChatForSupport": false,
    "canViewAllJobs": true,
    "canCreateUser": true,
    "canCreateRole": true,
    "canCreateSource": true,
    "canUploadFile": true,
    "canManageNodeActivity": true,
    "canManageEngines": true,
    "canManageQueues": true,
    "canManageEngineRouting": true,
    "canManageSupportSettings": true
  },
  "userCreatedAt": 1681311939789
}
```

note

If your password includes single or double quotes, you may need to escape the quotes in your authentication token request. The required escapes vary depending on how you send the request. For example, if you use cURL and the password is ``example'6852``, the password value should be ``example\'"6852\'`` in the authentication token request.

Use the ``token`` attribute of the JSON return object as a Bearer token, or append the prefix ``_dremio`` to compose a self-contained token as ``_dremio<tokenstring>``.

Example: Request Using A Self-Contained Authorization Token

```
curl -X GET 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: _dremio4ksrt534vk7fkq64xh55g7776b' \
--header 'Content-Type: application/json'
```

Errors

Error messages will be sent back in the response body using the following format:

Error Messages Format

```
{
  "errorMessage": "brief error message",
  "moreInfo": "detailed error message"
}
```

Query Parameters

Dremio supports query parameters for many API endpoints. The documentation for each API lists the supported query parameters for specific endpoints, along with any default and maximum values for the query parameters for that endpoint.

pageToken Query Parameter

Use the `pageToken` query parameter to split large sets of results into multiple pages.

Endpoints may support the `pageToken` query parameter based on either a built-in maximum number of results per page or a user-specified maximum that is established with a separate query parameter. The documentation for each API lists the built-in maximum or the query parameter to use to specify a maximum, as applicable.

note

Do not change any other query parameters included in the request URL when you use `pageToken`.

Built-in Maximum

If the endpoint has a built-in maximum number of results per page, responses automatically include a page token attribute when the response contains more results than the built-in maximum. Use the value for this token in the request URL as the `pageToken` value to retrieve the next page of results.

As an example, the Reflection summary endpoint supports the `pageToken` parameter. If the Reflection summary contains more than 50 results, the response will include the `nextPageToken` attribute. To retrieve the next 50 results, add `?pageToken={nextPageToken_value}` to the request URL:

Example Request with pageToken Query Parameter

```
curl -X GET \
  'https://{hostname}/api/v3/reflection-summary?pageToken=BhQxNjc0MjhlYi030TM2LTRlYTItYTTFmYi0yM2IxYWM2ZTk0NTQSA== ' \
  --header 'Authorization: Bearer <dremioAccessToken>' \
  --header 'Content-Type: application/json'
```

For subsequent requests, replace the `pageToken` value in the request URL with the token value from the previous response. If the response does not include a token attribute and value, you have retrieved the last page of available results.

User-Specified Maximum

For endpoints that require users to specify a maximum number of results per page with a separate query parameter, responses only include a page token attribute if your initial request URL includes the separate query parameter and the response contains more results than the maximum you specify. Add the value for this token to the request URL as the `pageToken` value, keeping the separate query parameter as well, to retrieve the next page of results.

Catalog API endpoints for retrieving non-filesystem [sources](#), [spaces](#), and [folders](#) by ID or path support the `maxChildren` query parameter for specifying the maximum number of child objects to include in each response. If the response contains more than the specified number of child objects, the response includes the `nextPageToken` attribute. To retrieve the next page of results, add `?pageToken={nextPageToken_value}` to the request URL. This example shows a request URL that uses the `nextPageToken` query parameter with the `maxChildren` query parameter set to 25:

Example Request with maxChildren and pageToken Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4?maxChildren=25&pageToken=BhQxNjc0MjhlYi030TM2LTRlYTIyYTFmYi0yM2IxYWM2ZTk0NTQSA==' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

For subsequent requests, replace the `pageToken` value in the request URL with the token value from the previous response. If the response does not include a token attribute and value, you have retrieved the last page of available results.

maxResults Query Parameter

Use the `maxResults` query parameter to specify the maximum number of results to retrieve in each request.

For example, if you want to retrieve no more than 25 results for an endpoint that supports the `maxResults` query parameter, append `?maxResults=25` to the request URL:

Example Request with maxResults Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?maxResults=25' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

filter Query Parameter

Use the `filter` query parameter to filter responses so that they include only results

with the specified attributes and values. The value for the `filter` query parameter is a URL-encoded JSON string that represents a JSON object that specifies the desired attributes and values.

As an example, the [Reflection summary](#) endpoint supports the `filter` query parameter for certain specific attributes. To retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths, the `filter` JSON object would look like this:

Example JSON Object for Filter

```
{
  "reflectionType": ["RAW"],
  "refreshStatus": ["MANUAL", "SCHEDULED"],
  "enabledFlag": true,
  "reflectionNameOrDatasetPath": "samples.dremio.com"
}
```

To use the JSON object as the `filter` value, convert it to URL-encoded JSON and add it to the request URL:

Example Request with filter Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the endpoint-specific documentation to learn which attributes each endpoint supports for the `filter` query parameter.

orderBy Query Parameter

Use the `orderBy` query parameter to organize the response in ascending or descending order based on the value of the specified attribute. The default is ascending order. To specify descending order, add a `-` character before the attribute name.

For example, the [Reflection summary](#) endpoint supports ordering the response by ReflectionName, datasetName, or reflectionType. To organize the response in ascending order by ReflectionName:

Example Request with orderBy Query Parameter (Ascending Order)

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

To organize the response in descending order, add a `-` before the attribute name:

Example Request with orderBy Query Parameter (Descending Order)

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=-reflectionName' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Read the endpoint-specific documentation to learn which attributes each endpoint supports for the `orderBy` query parameter.

limit and offset Query Parameters

The `limit` query parameter allows you to retrieve a specific number of results. For endpoints that support the `limit` query parameter, you can specify the number of results to retrieve. For example, if you only want to retrieve the first 10 available results, add `?limit=10` to the request URL:

Example Request for First 10 Results with Limit Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=10' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

The `offset` query parameter allows you to skip a specific number of results in the response. When a response contains many results, you can use the `limit` and `offset` query parameters together to break the response into pages.

For example, consider a job result response object that contains 5000 results. The Job API allows you to retrieve a maximum of 500 results per request. To retrieve all 5000 results, start by adding `?limit=500` to the request URL to retrieve the first 500:

Example Request for First 500 Results with Limit Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

In the next request, to retrieve the next 500 results (rows 501-1000), add `&offset=500` to the request URL:

Example Request for Results 501-1000 with limit and offset Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500&offset=500' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

To retrieve the next 500 results (rows 1001-1500), increment the `offset` parameter to 1000 in the next request:

Example Request for Results 1001-1500 with limit and offset Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/job/{id}/results?limit=500&offset=1000' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

```
--header 'Content-Type: application/json'
```

Continue incrementing the ``offset`` parameter in requests until you have retrieved all 5000 results.

Read the documentation for each API to learn about endpoint-specific support for the ``limit`` and ``offset`` query parameters.

type Query Parameter

Use the ``type`` query parameter to limit your request so that the response includes only results for the type you specify.

For example, if an endpoint supports the ``type`` query parameter, and the endpoint's list of valid values includes ``SOURCE``, you can limit the response so that it includes only results for sources. Append ``?type=SOURCE`` to the request URL:

Example Request with type Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/privileges?type=SOURCE' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the ``type`` query parameter, including lists of valid values.

include and exclude Query Parameters

Some APIs exclude non-default attributes or include lengthy attributes in the default GET responses. These APIs support the ``include`` and ``exclude`` query parameters, which you can use to include or exclude certain attributes in the responses for GET endpoints.

The ``include`` query parameter allows you to include non-default attributes in the response. For example, in the Catalog API, you can include a catalog object's ``permissions`` array in the response:

Example Request with include Query Parameter

```
curl -X GET \  
'https://{hostname}/api/v3/catalog/ffbe8c1d-1db7-48d1-9c58-f452838fedc0?include=permissions' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

The ``exclude`` query parameter allows you to exclude supported attributes from the response. For example, this Catalog API request excludes the object's ``children`` attribute from the response:

Example Request with exclude Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/ffbe8c1d-1db7-48d1-9c58-f452838fedc0?exclude=children' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `include` and `exclude` query parameters, including information about supported values.

createdBy Query Parameter

Use the `createdBy` query parameter to limit the request to objects created by a specific user.

For example, this Scripts API request retrieves only scripts that were created by the user whose ID is `8be516f3-04c4-4d19-824d-5a70b3c4442e`:

Example Request with createdBy Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?createdBy=8be516f3-04c4-4d19-824d-5a70b3c4442e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

ownedBy Query Parameter

Use the `ownedBy` query parameter to limit the request to objects owned by a specific user.

For example, this Scripts API request retrieves only scripts that are owned by the user whose ID is `8be516f3-04c4-4d19-824d-5a70b3c4442e`:

Example Request with ownedBy Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?ownedBy=8be516f3-04c4-4d19-824d-5a70b3c4442e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

search Query Parameter

Use the `search` query parameter to limit the request to objects that contain values that include the search string.

For example, the Scripts API supports the `search` query parameter for the name attribute. This Scripts API request retrieves only scripts whose values for the name attribute include `dev`:

Example Request with search Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/scripts?search=dev' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Read the documentation for each API to learn about endpoint-specific support for the `search` query parameter, including information about searchable attributes.

maxChildren Query Parameter

The `maxChildren` query parameter allows you to specify the maximum number of child objects to include in each response. This example shows a request URL that uses the `nextPageToken` query parameter with the `maxChildren` query parameter set to 25:

Example Request with maxChildren Query Parameter

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4?maxChildren=25' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Use the `maxChildren` query parameter in concert with the pageToken query parameter to split large sets of results into multiple pages.

Response Headers

Dremio API responses include HTTP headers that provide additional information about responses. Each header includes a case-insensitive name and a value, separated by a colon.

The following response headers are common to Dremio API endpoints:

Header Name	Example Value	Description
---	---	---
Allow	GET,OPTIONS	Request methods the endpoint supports.
Cache-Control	no-cache, no-store	Caching instructions for browsers and shared caches.
Content-Length	2188	Size of the response body, in bytes.
Content-Type	application/json	MIME type of the object.
Date	Fri, 14 Apr 2023 19:39:53 GMT	Date and time when the response originated.
Vary	Accept-Encoding, User-Agent	Names of request headers that could have affected the response's generation.
x-content-type-options	nosniff	Instructions about following the MIME type in the content-type header. Blocks content sniffing.
x-xss-protection	1; mode=block	Instructions used to stop pages from loading when a browser detects reflected cross-site scripting attacks.

Was this page helpful?

[\[Previous](#)

[Admin CLI\]\(/current/reference/admin-cli/\)\[Next](#)

[Catalog\]\(/current/reference/api/catalog/\)](#)

Base URL

Authentication

OAuth Access Tokens Enterprise

Personal Access Tokens Enterprise

Dremio Authentication Tokens

Example

Errors

Query Parameters

pageToken Query Parameter

maxResults Query Parameter

filter Query Parameter

orderBy Query Parameter

limit and offset Query Parameters

type Query Parameter

include and exclude Query Parameters

createdBy Query Parameter

ownedBy Query Parameter

search Query Parameter

maxChildren Query Parameter

Response Headers

Source:
[https://docs.dremio.com/current/reference/api/c
atalog/](https://docs.dremio.com/current/reference/api/catalog/)

Version: current [26.x]

On this page

Catalog

Use the Catalog API to retrieve and manage [sources](#) and [spaces](#) as well as the [folders](#),

files, tables, and views they contain. The Catalog API also includes endpoints for retrieving lineage information for datasets and for creating and managing tags, wikis, privileges, and grants on catalog objects.

Use the Catalog API endpoint described on this page to retrieve a list of the spaces and sources in your Dremio organization. The response contains the IDs required to make requests to other Catalog API endpoints to create, retrieve, update, and delete objects in your catalog.

Catalog Object

```
{
  "data": [
    {
      "id": "a7b1bc39-bffa-4c30-a5eb-5bdaf5bd0959",
      "path": [
        "@dremio"
      ],
      "tag": "0QVA7wGyiY0=",
      "type": "CONTAINER",
      "containerType": "HOME",
      "stats": {
        "datasetCount": 18,
        "datasetCountBounded": false
      },
      "permissions": [
        "READ",
        "WRITE",
        "ALTER_REFLECTION",
        "SELECT",
        "ALTER",
        "VIEW_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "CREATE_TABLE",
        "DROP",
        "EXTERNAL_QUERY",
        "INSERT",
        "TRUNCATE",
        "DELETE",
        "UPDATE",
        "EXECUTE",
        "CREATE_SOURCE",
        "ALL"
      ]
    },
    {
      "id": "ed1013cb-4fea-6552-8d43-015215a38bcc",
      "path": [
        "Testing"
      ],
      "tag": "PR1M7B1Rhjs=",
      "type": "CONTAINER",
      "containerType": "SPACE",
      "stats": {
```

```

    "datasetCount": 3,
    "datasetCountBounded": false
  },
  "createdAt": "2023-02-14T19:28:40.840Z",
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ]
},
{
  "id": "6b714877-760e-115b-aefd-799430b3ceab",
  "path": [
    "Samples"
  ],
  "tag": "nEjWZGnrA00=",
  "type": "CONTAINER",
  "containerType": "SOURCE",
  "stats": {
    "datasetCount": 10,
    "datasetCountBounded": false
  },
  "createdAt": "2023-01-04T22:13:02.536Z",
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",

```

```

    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ]
}
]
}

```

Catalog Attributes

data Array of Object

List of catalog objects in the Dremio organization.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the catalog object.

Example: ed1013cb-4fea-6552-8d43-015215a38bcc

path Array of String

Path of the catalog object within Dremio, expressed as an array.

Example: ["Testing"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the catalog object.

Example: PR1M7B1Rhjs=

type String

Type of the catalog object. For objects that can contain other catalog objects (the only objects this endpoint retrieves), the type is `CONTAINER`.

Example: CONTAINER

containerType String

For catalog objects with the type CONTAINER, the type of container.

Enum: SPACE, SOURCE, FOLDER, HOME

Example: SPACE

stats Object

Information about the number of datasets in the catalog object and whether the dataset

count is bounded. Appears in the response only if the request URL includes the `datasetCount` query parameter.

Example: `{"datasetCount": 18,"datasetCountBounded": false}`

createdAt String

Date and time that the catalog object was created, in UTC format.

Example: `2023-02-14T19:28:40.840Z`

permissions Array of String

Enterprise-only. List of the privileges that you have on the catalog object. Only appears in the response if the request URL includes the ``permissions`` query parameter. For more information, read [Privileges](#).

Example:

```
["READ","WRITE","ALTER\_REFLECTION","SELECT","ALTER","VIEW\_REFLECTION","MODIFY","MANAGE\_GRANTS","CREATE\_TABLE","DROP","EXTERNAL\_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE\_SOURCE","ALL"]
```

Attributes of the ``stats`` Object

datasetCount Integer

Number of datasets the catalog object contains.

Example: `18`

datasetCountBounded Boolean

If the dataset count is bounded, the value is ``true``. Otherwise, the value is ``false``.

Example: `false`

Retrieve a Catalog

Retrieve the catalog for the current Dremio instance.

Method and URL

```
GET /api/v3/catalog
```

Parameters

include Query String Optional

Include a non-default attribute in the response. The available values for the include query parameter are ``permissions`` (Enterprise-only) and ``datasetCount``. Specify ``permissions`` to include each catalog object's permissions array in the response. Specify ``datasetCount`` to include the [stats object](#) in the response. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
{  
  "data": [  
    {  
      "id": "a7b1bc39-bffa-4c30-a5eb-5bdaf5bd0959",  
      "path": [  
        "@dremio"  
      ],  
      "tag": "0QVA7wGyiY0=",  
      "type": "CONTAINER",  
      "containerType": "HOME"  
    },  
    {  
      "id": "ed1013cb-4fea-6552-8d43-015215a38bcc",  
      "path": [  
        "Testing"  
      ],  
      "tag": "PR1M7B1Rhjs=",  
      "type": "CONTAINER",  
      "containerType": "SPACE",  
      "createdAt": "2023-02-14T19:28:40.840Z"  
    },  
    {  
      "id": "6b714877-760e-115b-aefd-799430b3ceab",  
      "path": [  
        "Samples"  
      ],  
      "tag": "nEjWZGnrA00=",  
      "type": "CONTAINER",  
      "containerType": "SOURCE",  
      "createdAt": "2023-01-04T22:13:02.536Z"  
    }  
  ]  
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

API Reference](/current/reference/api/)[Next

Source](/current/reference/api/catalog/source/)

Catalog Attributes

Retrieve a Catalog

Parameters

Example

Response Status Codes

Source:
[https://docs.dremio.com/current/reference/api/d](https://docs.dremio.com/current/reference/api/data-maintenance)
ata-maintenance

Version: current [26.x]

On this page

Data Maintenance

Use the Data Maintenance API to manage and execute automatic maintenance tasks in the Open Catalog, including `OPTIMIZE` and `EXPIRE_SNAPSHOTS`. Enabled tasks run automatically based on Dremio logic.

TaskInfo Object

```
{
  "id": "a75918ee-431b-4296-a2b3-298dff53ab12",
  "type": "OPTIMIZE",
  "level": "CATALOG",
  "sourceName": "prod",
  "isEnabled": true,
  "config": {
    "tableId": "folder1.folder2.table1"
  }
}
```

TaskInfo Attributes

id String (UUID)

Task ID for this task.

Example: a75918ee-431b-4296-a2b3-298dff53ab12

type String

Type of maintenance to be run by this task.

``OPTIMIZE``: Run ``OPTIMIZE`` on one or more tables

``EXPIRE_SNAPSHOTS``: Run ``VACUUM EXPIRE SNAPSHOTS`` on one or more tables

Example: OPTIMIZE

level String

The scope of the maintenance task.

Supported values include:

``CATALOG``: Perform a maintenance task throughout the catalog

``TABLE``: Default, perform a maintenance task on one table

Example: CATALOG

sourceName String

The name of the Open Catalog.

Example: prod

isEnabled Boolean

Whether the task is enabled.

Example: true

config Object

Provides an object that contains the fully-qualified name of the target table of the maintenance task.

Example:

```
{
  "tableId": "folder1.folder2.table1"
}
```

List Tasks

Lists tasks with filtering, paging, and max results.

Method and URL

```
GET /api/v3/catalog/{sourceId}/maintenance/tasks?filter
```

Parameters

sourceId Path String

Identifier of the Open Catalog hosting the maintenance tasks.

Example: 186665c9-2a6a-46b7-9a7d-8d3a867ad650

filter Query String

A Common Expression Language (CEL) expression. For more information, see the [intro to CEL](#)

Usable variables within the expression are:

`type`: Optimization type. This can be OPTIMIZE or EXPIRE_SNAPSHOTS.

`level`: Maintenance level. This can be TABLE or CATALOG.

`tableId`: Table name preceded by zero or more namespaces(without a source name), with `.` as the separator.

Example: type=="OPTIMIZE"&&level=="TABLE"

maxResults Query Integer

Maximum number of results to be returned by the server. The server may return fewer resources than this, but not more.

Defaults to 10.

Example: maxResults=25

pageToken Query String

Opaque pagination token. This token is returned by the server in a previous response, and is used to request the next page of results. For example: 1->AAAAAQ, 2->AAAAAg, 3->AAAAAw, etc.

Example: pageToken=AAAAAQ

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/186665c9-2a6a-46b7-9a7d-8d3a867ad650/maintenance/tasks?filter=type=="OPTIMIZE"&&level=="TABLE"&maxResults=10' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "id": "d510279d-5741-4feb-82b8-d6923c9ac6c8",
```

```

    "type": "OPTIMIZE",
    "level": "TABLE",
    "sourceName": "prod",
    "isEnabled": true,
    "config": {
      "tableId": "folder1.folder2.table1"
    }
  },
  {
    "id": "b0fb84d0-b2c9-455d-9013-3c3e417c46f0",
    "type": "OPTIMIZE",
    "level": "TABLE",
    "sourceName": "prod",
    "isEnabled": true,
    "config": {
      "tableId": "folder1.folder2.table2"
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Create a Task

Creates a maintenance task.

Method and URL

```
POST /api/v3/catalog/{sourceId}/maintenance/tasks
```

Parameters

sourceId Path String

The sourceID of the Open Catalog performing this task.

Example: 186665c9-2a6a-46b7-9a7d-8d3a867ad650

taskDefinition Body Object

Represents a new task.

Example:

```
{
  "type": "OPTIMIZE",
  "isEnabled": true,
  "config": {
    "tableId": "folder1.folder2.table1"
  }
}
```

TaskDefinition Object Parameters

type String

Type of maintenance to run.

OPTIMIZE - A table optimization task

EXPIRE_SNAPSHOTS - A table vacuum task

isEnabled Boolean

Indicates if the task is enabled.

Example: true

config Body Object

Provides an object that contains a fully-qualified object name in the indicated catalog as as the target for the maintenance task.

Example:

```
{
  "tableId": "folder1.folder2.table1"
}
```

Example

Request

```
curl -X POST
'https://{hostname}/api/v3/catalog/186665c9-2a6a-46b7-9a7d-8d3a867ad650/maintenance/task
s' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
```

```
"type": "OPTIMIZE",
"isEnabled": true,
"config": {
  "tableId": "folder1.folder2.table1"
}
}'
```

Response

```
{
  "id": "f526e534-f61c-43a5-9d35-51a8b0f75826",
  "type": "OPTIMIZE",
  "level": "TABLE",
  "sourceName": "prod",
  "isEnabled": true,
  "config": {
    "tableId": "folder1.folder2.table1"
  }
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Retrieve a Task by ID

Retrieve a task by specifying the task's ID.

Method and URL

```
GET /api/v3/catalog/{sourceId}/maintenance/tasks/{taskId}
```

Parameters

sourceId Path String

Unique identifier of the Open Catalog hosting the task.

Example: 186665c9-2a6a-46b7-9a7d-8d3a867ad650

taskId Path String (UUID)

Unique identifier of the task that you want to retrieve.

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/186665c9-2a6a-46b7-9a7d-8d3a867ad650/maintenance/tasks/f526e534-f61c-43a5-9d35-51a8b0f75826' \
--header 'Authorization: Bearer <dremioAccessToken>' \
>>>>>> origin/rel/26.1
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "f526e534-f61c-43a5-9d35-51a8b0f75826",
  "type": "OPTIMIZE",
  "level": "TABLE",
  "sourceName": "prod",
  "isEnabled": true,
  "config": {
    "tableId": "folder1.folder2.table1"
  }
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Update a Task

Update the specified task, such as disabling it.

Method and URL

PUT /api/v3/catalog/{sourceId}/maintenance/tasks/{taskId}

Parameters

sourceId Path String

Unique identifier of the Open Catalog hosting the task.

Example: 186665c9-2a6a-46b7-9a7d-8d3a867ad650

taskId Path String (UUID)

Unique identifier of the task to update

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

taskDefinition Body Object

Represents an updated task specification.

Example:

```
{
  "type": "OPTIMIZE",
  "isEnabled": false,
  "config": {
    "tableId": "folder1.folder2.table1"
  }
}
```

Example

Request

```
curl -X PUT \
'https://{hostname}/api/v3/catalog/186665c9-2a6a-46b7-9a7d-8d3a867ad650/maintenance/task/f526e534-f61c-43a5-9d35-51a8b0f75826' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "type": "OPTIMIZE",
  "isEnabled": false,
  "config": {
    "tableId": "folder1.folder2.table1"
  }
}'
```

Response

```
{
```

```
"id": "f526e534-f61c-43a5-9d35-51a8b0f75826",
"type": "OPTIMIZE",
"level": "TABLE",
"sourceName": "prod",
"isEnabled": false,
"config": {
  "tableId": "folder1.folder2.table1"
}
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Delete a Task

Delete the specified task.

Method and URL

```
DELETE /api/v3/catalog/{sourceId}/maintenance/tasks/{taskId}
```

Parameters

sourceId Path String

Unique identifier of the source from where to retrieve the task.

Example: 186665c9-2a6a-46b7-9a7d-8d3a867ad650

taskId Path String (UUID)

Unique identifier of the task that you want to retrieve.

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/186665c9-2a6a-46b7-9a7d-8d3a867ad650/maintenance/tasks/f526e534-f61c-43a5-9d35-51a8b0f75826' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

1

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Was this page helpful?

[Previous

Grants](/current/reference/api/catalog/grants)[Next

Dataset](/current/reference/api/datasets/)

TaskInfo Attributes

List Tasks

Parameters

Example

Response Status Codes

Create a Task

Parameters

Example

Response Status Codes

Retrieve a Task by ID

Parameters

Example

Response Status Codes

Update a Task

Parameters

Example

Response Status Codes

Delete a Task

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/datasets/>

Version: current [26.x]

On this page

Dataset

note

The Dataset API is supported in Dremio 25.0.5+.

Use the Dataset API to retrieve Dremio's Reflection recommendations for your datasets.

Dataset Object (All Reflections)

```
{
  "data": [
    {
      "type": "RAW",
      "enabled": true,
      "arrowCachingEnabled": false,
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {

```

```

        "name": "fare_amount"
    },
    {
        "name": "tip_amount"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "dropoff_date"
    },
    {
        "name": "passenger_count"
    }
],
"entityType": "reflection"
},
{
    "type": "AGGREGATION",
    "enabled": true,
    "arrowCachingEnabled": false,
    "dimensionFields": [
        {
            "name": "passenger_count",
            "granularity": "DATE"
        }
    ],
    "measureFields": [
        {
            "name": "total_amount",
            "measureTypeList": [
                "COUNT",
                "SUM"
            ]
        },
        {
            "name": "trip_distance_mi",
            "measureTypeList": [
                "COUNT",
                "SUM"
            ]
        },
        {
            "name": "fare_amount",
            "measureTypeList": [
                "COUNT",
                "SUM"
            ]
        },
        {
            "name": "tip_amount",
            "measureTypeList": [

```

```

        "COUNT",
        "SUM"
    ]
  },
],
"partitionFields": [
  {
    "name": "dropoff_date"
  },
  {
    "name": "passenger_count"
  }
],
"entityType": "reflection"
}
]
}

```

Dataset Attributes

data Array of Object

List of recommended Reflection objects for the specified dataset ID.

Attributes of objects in the `data` Array

type String

Reflection type. For details, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: RAW

enabled Boolean

If the Reflection is available for accelerating queries, `true`. Otherwise, `false`.

Example: true

arrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, `true`. Otherwise, `false`.

Example: false

displayFields Array of Object

Information about the fields displayed from the anchor dataset. Each object in the displayFields array contains one attribute: name. Included only for raw Reflections. Not included for aggregation Reflections.

Example:

```
[{"name": "pickup_datetime"}, {"name": "passenger_count"}, {"name": "trip_distance_mi"}, {"name": "fare_amount"}, {"name": "tip_amount"}, {"name": "total_amount"}]
```

dimensionFields Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each object in the dimensionFields array contains two attributes: name and granularity. Included only for aggregation Reflections. If the anchor dataset does not include any dimension fields, the dimensionFields value is an empty array. Not included for raw Reflections.

Example: [{"name": "passenger_count", "granularity": "DATE"}]

measureFields Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each object in the measureFields array contains two attributes: name and measureTypeList. Included only for aggregation Reflections. If the anchor dataset does not include any measure fields, the measureFields value is an empty array. Not included for raw Reflections.

Example:

```
[{"name": "total_amount", "measureTypeList": ["COUNT", "SUM"]}, {"name": "trip_distance_mi", "measureTypeList": ["COUNT", "SUM"]}, {"name": "fare_amount", "measureTypeList": ["COUNT", "SUM"]}, {"name": "tip_amount", "measureTypeList": ["COUNT", "SUM"]}]
```

partitionFields Array of Object

Information about the fields from the anchor dataset used to partition data in the Reflection. Each object in the partitionFields array contains one attribute: name. Included only for aggregation Reflections. If the anchor dataset does not include any partition fields, the partitionFields value is an empty array. Not included for raw Reflections.

Example: [{"name": "dropoff_date"}, {"name": "passenger_count"}]

entityType String

Type of entity. For objects in dataset responses, the entityType is `reflection`.

Create and Retrieving Reflection Recommendations for a Dataset

Create Reflection recommendations for the specified dataset. The response contains the Reflection recommendations.

Method and URL

```
POST /api/v3/dataset/{id}/reflection/recommendation/{type}/
```

Parameters

id Path String (UUID)

The id of the dataset for which you want to create and retrieve recommended Reflections.

Example: 88e5fbdf-4b56-4286-9b8b-bb48e1f350eb

type Path String

The type of Reflection recommendations you want to create and retrieve.

ALL: Create and retrieve both raw and aggregation Reflection recommendations.

RAW: Create and retrieve only raw Reflection recommendations.

AGG: Create and retrieve only aggregation Reflection recommendations.

NOTE: The type is not case-sensitive. For example, `AGG`, `agg`, and `aGg` are valid type values for aggregation Reflection recommendations.

Example: ALL

Example Request (All Reflections)

```
curl -X POST 'https://{hostname}/api/v3/dataset/88e5fbdf-4b56-4286-9b8b-bb48e1f350eb/reflection/recommendation/ALL/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response (All Reflections)

```
{
  "data": [
    {
      "type": "RAW",
      "enabled": true,
      "arrowCachingEnabled": false,
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {
          "name": "fare_amount"
        },
        {
          "name": "tip_amount"
        },
        {
          "name": "total_amount"
        }
      ]
    }
  ]
}
```

```

    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "entityType": "reflection"
},
{
  "type": "AGGREGATION",
  "enabled": true,
  "arrowCachingEnabled": false,
  "dimensionFields": [
    {
      "name": "passenger_count",
      "granularity": "DATE"
    }
  ],
  "measureFields": [
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionFields": [
    {

```

```

        "name": "dropoff_date"
      },
      {
        "name": "passenger_count"
      }
    ],
    "entityType": "reflection"
  }
]
}

```

Example Request (Raw Reflections)

```

curl -X POST \
  'https://{hostname}/api/v3/dataset/88e5fbdf-4b56-4286-9b8b-bb48e1f350eb/reflection/recom
mendation/RAW/' \
  --header 'Authorization: Bearer <dremioAccessToken>' \
  --header 'Content-Type: application/json'

```

Example Response (Raw Reflections)

```

{
  "data": [
    {
      "type": "RAW",
      "enabled": true,
      "arrowCachingEnabled": false,
      "displayFields": [
        {
          "name": "pickup_datetime"
        },
        {
          "name": "passenger_count"
        },
        {
          "name": "trip_distance_mi"
        },
        {
          "name": "fare_amount"
        },
        {
          "name": "tip_amount"
        },
        {
          "name": "total_amount"
        }
      ],
      "partitionFields": [
        {
          "name": "dropoff_date"
        },
        {

```

```

        "name": "passenger_count"
      }
    ],
    "entityType": "reflection"
  }
]
}

```

Example Request (Aggregation Reflections)

```

curl -X POST 'https://{hostname}/api/v3/dataset/88e5fbdf-4b56-4286-9b8b-bb48e1f350eb/reflection/recommendation/AGG/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'

```

Example Response (Aggregation Reflections)

```

{
  "data": [
    {
      "type": "AGGREGATION",
      "enabled": true,
      "arrowCachingEnabled": false,
      "dimensionFields": [
        {
          "name": "passenger_count",
          "granularity": "DATE"
        }
      ],
      "measureFields": [
        {
          "name": "total_amount",
          "measureTypeList": [
            "COUNT",
            "SUM"
          ]
        },
        {
          "name": "trip_distance_mi",
          "measureTypeList": [
            "COUNT",
            "SUM"
          ]
        },
        {
          "name": "fare_amount",
          "measureTypeList": [
            "COUNT",
            "SUM"
          ]
        }
      ]
    }
  ]
}

```

```

    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "entityType": "reflection"
}
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

405 Method Not Allowed

500 Internal Server Error

Was this page helpful?

[Previous

Catalog][(/current/reference/api/catalog/)] [Next

Engine Management][(/current/reference/api/engine-management/)]

Dataset Attributes

Create and Retrieving Reflection Recommendations for a Dataset

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/engine-management/>

Version: current [26.x]

On this page

Engine Management Enterprise

Use the Engine Management API to manage your engines in Dremio:

Engine Management - Covers the objects and operations for managing engines.

****Engine Configuration**** - Covers the objects and operations for engine configuration options.

Engine Object

```
{
  "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
  "name": "my-engine",
  "version": "xNMoJiZAyfA=",
  "spec": {
    "size": "2XSmall",
    "resourceAllocationOffset": "reserve-2-4",
    "targetCpuCapacity": "32C",
    "spillStorageSize": "128GB",
    "c3StorageSize": "128GB",
    "idleTimeout": "PT1H",
    "executorPodMetadata": {
      "labels": {
        "environment": "production",
        "team": "analytics"
      },
      "annotations": {
        "key1": "value1",
        "key2": "value2"
      },
      "nodeSelectors": {
        "kubernetes.io/hostname": "node-1"
      },
      "tolerations": [
        {
          "key": "example-key",
          "operator": "Exists",
          "effect": "NoExecute",
          "tolerationSeconds": 3600
        }
      ]
    },
    "status": {
      "state": "RUNNING",
      "message": "",
      "runningReplicas": 1,
      "startingReplicas": 0,
      "stoppingReplicas": 0,
      "runningExecutors": 1,
```

```

"startingExecutors": 0,
"stoppingExecutors": 0,
"replicas": [
  {
    "id": "000",
    "executors": [
      {
        "name": "executor-1",
        "memory": 10.45,
        "cpu": 5.33,
        "status": "RUNNING",
        "details": "No issues detected"
      }
    ]
  }
]
}
}
}

```

Engine Attributes

id String (UUID)

Unique identifier for the engine, generated by Dremio.

Example: d480b906-6a2d-486e-a1f6-a2e86452eab8

name String

Name of the engine.

The name of the engine must follow these rules:

Must start with a lowercase alphanumeric character (`[a-z0-9]`).

Must end with a lowercase alphanumeric character (`[a-z0-9]`).

Must contain only lowercase alphanumeric characters or a hyphen (`[\-a-z0-9]`).

Must be under 30 characters in length.

Must be unique and not previously used for any existing or deleted engines.

Example: my-engine

version String

A tag that identifies the version of the resource.

When modifying a resource, you will need to send this tag to the server. It will reject the modification request if the provided version does not match the server's version, as this indicates that the resource has been modified since it was last read.

Example: xNMojIZAyfA=

spec Object

Information about the engine specification.

Example: { "c3StorageSize": "128GB", "executorPodMetadata": { "annotations": { "key1": "value1", "key2": "value2" }, "labels": { "environment": "production", "team": "analytics" }, "nodeSelectors": { "kubernetes.io/hostname": "node-1" }, "tolerations": [{ "key": "example-key", "operator": "Exists", "effect": "NoExecute", "tolerationSeconds": 3600 }] }, "idleTimeout": "PT1H", "resourceAllocationOffset": "reserve-2-4", "size": "2XSmall", "spillStorageSize": "128GB", "targetCpuCapacity": "32C"},

status Object

Information about the engine status.

Example: { "state": "RUNNING", "message": "Engine is running", "runningReplicas": 2, "startingReplicas": 0, "stoppingReplicas": 0, "replicas": [{ "id": "001", "executors": [{ "name": "executor-1", "cpu": 5.33, "memory": 10.45, "status": "RUNNING", "details": "No issues detected" }, { "name": "executor-2", "cpu": 7.25, "memory": 12.67, "status": "RUNNING", "details": "Stable performance" }] }] }

Attributes of the `spec` Object

size String

T-Shirt size of the engine, which must be one of the values predefined in the [Engine Configuration](#).

Example: 2XSmall

resourceAllocationOffset String

Resource allocation offset level of the engine, which must be one of the values predefined in the [Engine Configuration](#).

Example: reserve-2-4

targetCpuCapacity String

CPU target of the engine, which must be one of the values predefined in the [Engine Configuration](#).

Example: 32C

spillStorageSize String

Disk size for the spill volume, which must be one of the values predefined in the [Engine Configuration](#).

Example: 128GB

c3StorageSize String

Disk size for the C3 cache, which must be one of the values predefined in the [Engine Configuration](#).

Example: 128GB

idleTimeout String

Amount of time before an engine stops automatically. The engine will also start automatically.

If absent, the engine must be started/stopped manually. If present but null, the system default will be used.

Specified as an ISO-8601 duration.

Example: PT1H

executorPodMetadata Object

Information about the Kubernetes metadata that can be applied to an executor pod.

Example: { "labels": { "environment": "production", "team": "analytics" }, "annotations": { "key1": "value1", "key2": "value2" }, "nodeSelectors": { "kubernetes.io/hostname": "node-1" }, "tolerations": [{ "key": "example-key", "operator": "Exists", "effect": "NoExecute", "tolerationSeconds": 3600 }] }

Attributes of the `executorPodMetadata` Object

labels Object

Kubernetes metadata labels to apply to executor pods.

A map of label keys (string) to label values (string).

For more information, refer to <https://kubernetes.io/docs/concepts/overview/working-with-objects/labels/>.

Example: { "environment": "production", "team": "analytics" }

annotations Object

Kubernetes metadata annotations to apply to executor pods.

A map of annotation keys (string) to annotation values (string).

For more information, refer to <https://kubernetes.io/docs/concepts/overview/working-with-objects/annotations/>.

Example: { "key1": "value1", "key2": "value2" }

nodeSelectors Object

Kubernetes node selectors to apply to executor pods.

A map of label keys (string) to label values (string).

For more information, refer to <https://kubernetes.io/docs/concepts/scheduling-eviction/assign-pod-node/>.

Example: { "kubernetes.io/hostname": "node-1" }

tolerations Array of Objects

Kubernetes tolerations to apply to executor pods.

For more information, refer to <https://kubernetes.io/docs/concepts/scheduling-eviction/taint-and-toleration/>.

Example: { "key": "example-key", "operator": "Exists", "effect": "NoExecute", "tolerationSeconds": 3600 }

Attributes of the `toleration` Object

key String

The toleration key.

Example: example-key

operator String

The toleration operator.

Enum: Exists, Equal

Example: Exists

effect String

The toleration effect.

Enum: NoSchedule, PreferNoSchedule, NoExecute

Example: NoExecute

tolerationSeconds Integer

The toleration seconds, to be used only with effect `NoExecute`.

Example: 3600

value Integer

The toleration value.

Must be empty if operator is equal to `Exists`.

Attributes of the `engineStatus` Object

state String

Current state of the engine.

Enum: STARTING, RUNNING, STOPPING, STOPPED, WARNING, FAILED

Example: STARTING

message String

Human readable explanation of why the current state was achieved, including error messaging.

Example: No replicas are running

runningReplicas Integer

Number of running replicas.

Example: 1

startingReplicas Integer

Number of replicas starting.

Example: 1

stoppingReplicas Integer

Number of replicas stopping.

Example: 0

runningExecutors Integer

Number of executors replicas.

Example: 1

startingExecutors Integer

Number of executors starting.

Example: 1

stoppingExecutors Integer

Number of executors stopping.

Example: 0

replicas Array of Object

Running replicas with resource usage.

```
Example: { "id": "000", "executors": [ { "name":  
"dremio-executor-my-engine-000-0.dremio-cluster-pod.dremio.svc.cluster.local",  
"memory": 0.11473152822395594, "cpu": 0.07142857142857142, "status": "RUNNING"  
} ] }
```

Attributes of Objects in the `replica` Array

id String

Unique identifier for the replica in the engine, generated by Dremio.

Example: 001

executors Array of Object

Running executors with resource usage.

```
Example: [ { "name": "executor-1", "cpu": 5.33, "memory": 10.45, "status": "RUNNING",  
"details": "No issues detected" }, { "name": "executor-2", "cpu": 7.25, "memory":  
12.67, "status": "RUNNING", "details": "Stable performance" } ]
```

Attributes of Objects in the `executor` Array

name String

Name of node. It's the hostname of the Kubernetes pod.

Example: dremio-master-0.dremio-cluster-pod.prod.svc.cluster.local

cpu Double

Percentage of CPU used over the last second for the replica.

Example: 5.33

memory Double

Percentage of direct memory in use for the replica.

Example: 10.45

status String

Status of the executor.

Enum: RUNNING, FAILED, WARNING

Example: RUNNING

details String

Error message sent by a running executor.

Example: Invalid version

Create an Engine

Create a new engine.

Method and URL

```
POST /api/v3/engines
```

Parameters

name Body String

Name of the engine.

The name of the engine must follow these rules:

Must start with a lowercase alphanumeric character (``[a-z0-9]``).

Must end with a lowercase alphanumeric character (``[a-z0-9]``).

Must contain only lowercase alphanumeric characters or a hyphen (``[\-a-z0-9]``).

Must be under 30 characters in length.

Must be unique and not previously used for any existing or deleted engines.

Example: my-engine

newEngineSpec Body Object

Represents a new engine specification, with the option to use system defaults.

Example: { "size": "2XSmall", "targetCpuCapacity": "16C", "spillStorageSize": "100GB", "c3StorageSize": "100GB", "idleTimeout": "PT2H" }

Parameters of the `newEngineSpec` Object

size String

T-shirt size of the engine.

Example: 2XSmall

targetCpuCapacity String

CPU target of the engine, which must be one of the values predefined in the [Engine Configuration](#).

Example: 32C

spillStorageSize String

Disk size for the spill volume, which must be one of the values predefined in the [Engine Configuration](#).

Example: 128GB

c3StorageSize String

Disk size for the C3 cache, which must be one of the values predefined in the [Engine Configuration](#).

Example: 128GB

idleTimeout String

Amount of time before an engine stops automatically. Engine will also start automatically. If unset, engine must be started/stopped manually. Specified as an [ISO-8601 duration](#).

Example: PT1H

executorPodMetadata Object

Information about the Kubernetes metadata that can be applied to an executor pod.

Example: { "labels": { "environment": "production", "team": "analytics" }, "annotations": { "key1": "value1", "key2": "value2" }, "nodeSelectors": { "kubernetes.io/hostname": "node-1" }, "tolerations": [{ "key": "example-key", "operator": "Exists", "effect": "NoExecute", "tolerationSeconds": 3600 }] }

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/engines' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--data-raw {
  "name": "my-engine",
  "spec": {
    "size": "2XSmall",
    "targetCpuCapacity": "16C",
    "spillStorageSize": "100GB",
    "c3StorageSize": "100GB",
    "idleTimeout": "PT2H"
  }
}
```

Response

```
{
  "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
  "name": "my-engine",
  "version": "FHj7/8hHG9U=",
  "spec": {
    "size": "2XSmall",
    "resourceAllocationOffset": "reserve-2-8",
    "targetCpuCapacity": "16C",
    "spillStorageSize": "100GB",
    "c3StorageSize": "100GB",
    "idleTimeout": "PT2H"
  },
  "status": {
    "state": "STARTING",
    "message": "",
    "runningReplicas": 0,
    "startingReplicas": 0,
    "stoppingReplicas": 0,
    "runningExecutors": 0,
    "startingExecutors": 0,
    "stoppingExecutors": 0,
    "replicas": []
  }
}
```

Response Status Codes

201 Engine created successfully

400 Bad Request

401 Unauthorized

403 Forbidden

List Engines

Retrieve the list of engines.

Method and URL

GET /api/v3/engines

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/engines' \
--header 'Authorization: Bearer <PersonalAccessToken>'
```

Response

```
{
  "data": [
    {
      "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
      "name": "my-engine",
      "version": "reg6Ll6mCH8=",
      "spec": {
        "size": "2XSmall",
        "resourceAllocationOffset": "reserve-2-8",
        "targetCpuCapacity": "16C",
        "spillStorageSize": "100GB",
        "c3StorageSize": "100GB",
        "idleTimeout": "PT2H"
      },
      "status": {
        "state": "RUNNING",
        "message": "",
        "runningReplicas": 1,
        "startingReplicas": 0,
        "stoppingReplicas": 0,
        "runningExecutors": 1,
        "startingExecutors": 0,
        "stoppingExecutors": 0,
        "replicas": [
          {
            "id": "000",
            "executors": [
              {
                "name":
"dremio-executor-my-engine-000-0.dremio-cluster-pod.dremio.svc.cluster.local",
                "memory": 0.25926326023549745,
                "cpu": 0.0,
```

```
    "status": "RUNNING"
  }
]
}
}
}
}
```

Response Status Codes

200 List of engines

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve an Engine by ID

Retrieve an engine by specifying the engine's ID.

Method and URL

```
GET /api/v3/engines/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the engine that you want to retrieve.

Example: ec3fc60f-3b8a-4a26-bdaf-db97ae39c50d

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/engines/c3be6992-07d9-415d-89e4-571ceab8c51f' \
--header 'Authorization: Bearer <PersonalAccessToken>'
```

Response

```
{
  "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
```



```

"name": "my-engine",
"version": "reg6Ll6mCH8=",
"spec": {
  "size": "2XSmall",
  "resourceAllocationOffset": "reserve-2-8",
  "targetCpuCapacity": "16C",
  "spillStorageSize": "100GB",
  "c3StorageSize": "100GB",
  "idleTimeout": "PT2H"
},
"status": {
  "state": "RUNNING",
  "message": "",
  "runningReplicas": 1,
  "startingReplicas": 0,
  "stoppingReplicas": 0,
  "runningExecutors": 1,
  "startingExecutors": 0,
  "stoppingExecutors": 0,
  "replicas": [
    {
      "id": "000",
      "executors": [
        {
          "name":
"dremio-executor-my-engine-000-0.dremio-cluster-pod.dremio.svc.cluster.local",
          "memory": 0.25926326023549745,
          "cpu": 0.0,
          "status": "RUNNING"
        }
      ]
    }
  ]
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden

Update an Engine

Update the specified engine.

Method and URL

```
PUT /api/v3/engines/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the engine that you want to update.

Example: ec3fc60f-3b8a-4a26-bdaf-db97ae39c50d

version Body String

A tag that identifies the version of the resource that this modification is based on. You must obtain this tag by first getting the engine. The server will reject the request if the provided version does not match the server's version, as this indicates that the resource has been modified since it was last read.

Example: xNMojiZAYfA=

spec Body Object

Represents an updated engine specification, with the option to use system defaults.

Example: { "size": "2XSmall", "targetCpuCapacity": "16C", "spillStorageSize": "100GB", "c3StorageSize": "100GB", "idleTimeout": "PT2H" }

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/engines/c3be6992-07d9-415d-89e4-571ceab8c51f' \
--header 'Authorization: Bearer <PersonalAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "my-engine",
  "version": "neLlpVqs0dc=",
  "spec": {
    "size": "XSmall",
    "targetCpuCapacity": "16C",
    "spillStorageSize": "100GB",
    "c3StorageSize": "100GB",
    "idleTimeout": "PT1H"
  }
}'
```

Response

```
{
  "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
  "name": "my-engine",
  "version": "ZLdaCiHCsic=",
  "spec": {
```

```

    "size": "XSmall",
    "resourceAllocationOffset": "reserve-2-8",
    "targetCpuCapacity": "16C",
    "spillStorageSize": "100GB",
    "c3StorageSize": "100GB",
    "idleTimeout": "PT1H"
  },
  "status": {
    "state": "STOPPING",
    "message": "",
    "runningReplicas": 0,
    "startingReplicas": 0,
    "stoppingReplicas": 0,
    "runningExecutors": 0,
    "startingExecutors": 0,
    "stoppingExecutors": 0,
    "replicas": []
  }
}

```

Response Status Codes

- 200 Engine updated successfully
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Engine not found
- 409 Version mismatch. Please fetch the latest resource and try again.

Stop an Engine

Stop the specified engine.

Method and URL

```
PUT /api/v3/engines/{id}/stop
```

Parameters

id Path String (UUID)

Unique identifier of the engine that you want to stop.

Example: ec3fc60f-3b8a-4a26-bdaf-db97ae39c50d

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/engines/c3be6992-07d9-415d-89e4-571ceab8c51f/stop' \
--header 'Authorization: Bearer <PersonalAccessToken>'
```

Response

```
{
  "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
  "name": "my-engine",
  "version": "GdDM93EM8Tk=",
  "spec": {
    "size": "XSmall",
    "resourceAllocationOffset": "reserve-2-8",
    "targetCpuCapacity": "16C",
    "spillStorageSize": "100GB",
    "c3StorageSize": "100GB",
    "idleTimeout": "PT1H"
  },
  "status": {
    "state": "STOPPING",
    "message": "",
    "runningReplicas": 0,
    "startingReplicas": 0,
    "stoppingReplicas": 0,
    "runningExecutors": 0,
    "startingExecutors": 0,
    "stoppingExecutors": 0,
    "replicas": []
  }
}
```

Response Status Codes

200 Engine stopped successfully

401 Unauthorized

403 Forbidden

404 Engine not found

Start an Engine

Start the specified engine.

Method and URL

```
PUT /api/v3/engines/{id}/start
```

Parameters

id Path String (UUID)

Unique identifier of the engine that you want to start.

Example: ec3fc60f-3b8a-4a26-bdaf-db97ae39c50d

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/engines/c3be6992-07d9-415d-89e4-571ceab8c51f/start' \
--header 'Authorization: Bearer <PersonalAccessToken>'
```

Response

```
{
  "id": "c3be6992-07d9-415d-89e4-571ceab8c51f",
  "name": "my-engine",
  "version": "aGnFL1ok1/Y=",
  "spec": {
    "size": "XSmall",
    "resourceAllocationOffset": "reserve-2-8",
    "targetCpuCapacity": "16C",
    "spillStorageSize": "100GB",
    "c3StorageSize": "100GB",
    "idleTimeout": "PT1H"
  },
  "status": {
    "state": "STARTING",
    "message": "",
    "runningReplicas": 0,
    "startingReplicas": 0,
    "stoppingReplicas": 0,
    "runningExecutors": 0,
    "startingExecutors": 0,
    "stoppingExecutors": 0,
    "replicas": []
  }
}
```

Response Status Codes

200 Engine started successfully

401 Unauthorized

403 Forbidden

404 Engine not found

Delete an Engine

Delete the specified engine.

Method and URL

```
DELETE /api/v3/engines/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the engine that you want to start.

Example: ec3fc60f-3b8a-4a26-bdaf-db97ae39c50d

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/engines/c3be6992-07d9-415d-89e4-571ceab8c51f' \
--header 'Authorization: Bearer <PersonalAccessToken>'
```

Response

```
1
```

Response Status Codes

200 Engine deleted successfully

401 Unauthorized

403 Forbidden

404 Engine not found

Was this page helpful?

[Previous

Dataset](/current/reference/api/datasets/)[Next

Configuration

Options](/current/reference/api/engine-management/engine-configuration-options)

Engine Attributes

Create an Engine

Parameters

Example

Response Status Codes

List Engines

Example

Response Status Codes

Retrieve an Engine by ID

Parameters

Example

Response Status Codes

Update an Engine

Parameters

Example

Response Status Codes

Stop an Engine

Parameters

Example

Response Status Codes

Start an Engine

Parameters

Example

Response Status Codes

Delete an Engine

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/external-token-providers/>

Version: current [26.x]

On this page

External Token Providers Enterprise

Use the External Token Providers API to manage external token providers that enable client applications to use a JSON Web Token (JWT) issued by an identity provider to authenticate to Dremio.

External Token Providers Object

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "userClaim": "upn",
  "issuer":
    "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks":
    "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
  "type": "JWT",
  "state": "ENABLED"
}
```

External Token Providers Attributes

id String (UUID)

Unique identifier of the external token provider. Generated by Dremio and immutable.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

name String

User-provided name of the external token provider. Used for display only.

Example: My Token Provider

audience Array of String

Intended recipients of the JSON Web Token (JWT). If there is only one audience for the JWT, then the audience value contains only one string.

Example: ["f7fdd9e0-8332-4131-95ce-b350c3bbeab2"]

userClaim String

Key name for the target claim in the JSON Web Token (JWT). The target claim's value corresponds to the Dremio username.

Example: upn

issuer String

URL that identifies the principal that issued the JSON Web Token (JWT).

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0>

jwks String

Endpoint that hosts the JWK Set (JWKS), a set of public keys used to verify the JSON Web Token (JWT) signature.

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys>

type String

Type of the tokens associated with the external token provider. The type is always `JWT`.

state String

Current state of the external token provider: `ENABLED` (default) or `DISABLED`.

Example: ENABLED

List All External Token Providers

Retrieve a list of all available external token providers and the metadata for each provider.

Method and URL

```
GET /api/v3/external-token-providers/
```

Parameters

pageToken Query Query Optional

Token for retrieving the next page of external token providers. If the Dremio instance has more providers than the maximum per page (default 5), the response will include a nextPageToken after the data array. Use the nextPageToken value in your request URL as the pageToken value. Do not change any other query parameters included in the request URL when you use pageToken. Read pageToken Query Parameter for usage examples.

limit Query Integer Optional

Number of rows to return. Maximum valid value is `99`. Default is `5`. Read Limit and Offset Query Parameters for usage examples.

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/external-token-providers' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "id": "d21bbf72-6ab7-45e8-9466-ae1d9ffe92a7",
      "name": "My First Provider",
      "type": "JWT",
      "state": "DISABLED"
    },
    {
      "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
      "name": "My Token Provider",
      "type": "JWT",
      "state": "ENABLED"
    }
  ]
}
```

Response Status Codes

200 OK

403 Forbidden

Create an External Token Provider

Create an external token provider.

Method and URL

```
POST /api/v3/external-token-providers
```

Parameters

name Body String

Name to use for the external token provider.

Example: My Token Provider

audience Body Array of String

Intended recipients of the JSON Web Token (JWT).

Example: ["f7fdd9e0-8332-4131-95ce-b350c3bbeab2"]

userClaim Body String

Key name for the target claim in the JSON Web Token (JWT). The target claim's value corresponds to the Dremio username.

Example: upn

issuer Body String

URL that identifies the principal that issued the JSON Web Token (JWT).

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0>

jwt Body String Optional

Endpoint that hosts the JWK Set (JWKS), a set of public keys used to verify the JSON Web Token (JWT) signature. If you do not provide a jwt value, Dremio retrieves the value from `<issuer>/.well-known/openid-configuration`.

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys>

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/external-token-providers' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "issuer":
    "https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/v2.0",
  "jwt":
    "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
  "userClaim": "upn"
}'
```

Response

```
{
  "id": "a32191f2-ede6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
}
```

```

"audience": [
  "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
],
"userClaim": "upn",
"issuer":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
"jwks":
"https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
"type": "JWT",
"state": "ENABLED"
}

```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

Retrieve an External Token Provider by ID

Retrieve a specific external token provider by the providers's ID.

Method and URL

```
GET /api/v3/external-token-providers/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the external token provider.

Example:a32191f2-ed6-4533-9a17-1532eea015aa

Example

Request

```

curl -X GET
'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'

```

Response

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "f7fdd9e0-8332-4131-95ce-b350c3bbeab2"
  ],
  "userClaim": "upn",
  "issuer": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
  "type": "JWT",
  "state": "ENABLED"
}
```

Response Status Codes

200 OK

403 Forbidden

404 Not Found

Update an External Token Provider

Update the specified external token provider.

Method and URL

```
PUT /api/v3/external-token-providers/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the external token provider.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

name Body String

User-provided name of the external token provider.

Example: My Token Provider

audience Body Array of String

Intended recipients of the JSON Web Token (JWT). If there is only one audience for the JWT, then the audience value contains only one string.

Example: ["28edee01-4d0d-46ed-b1ae-52139bc3b3ad"]

userClaim Body String

Key name for the target claim in the JSON Web Token (JWT). The target claim's value corresponds to the Dremio username.

Example: preferred_username

issuer Body String

URL that identifies the principal that issued the JSON Web Token (JWT).

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0>

jwks Body String Optional

Endpoint that hosts the JWK Set (JWKS), a set of public keys used to verify the JSON Web Token (JWT) signature. If you do not provide a jwks value, Dremio retrieves the value from ``<issuer>/well-known/openid-configuration`` using the updated issuer.

Example:

<https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys>

type Body String Optional

Type of the tokens associated with the external token provider. The type is always `JWT`.

Example: JWT

state Body String Optional

Current state of the external token provider: `ENABLED` or `DISABLED`. If the update request does not include the state parameter, Dremio does not change the state.

Example: ENABLED

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "My Token Provider",
  "audience": [
    "28edee01-4d0d-46ed-b1ae-52139bc3b3ad"
  ],
  "userClaim": "preferred_username",
  "issuer": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0"
```

```
}'
```

Response

```
{
  "id": "a32191f2-ed6-4533-9a17-1532eea015aa",
  "name": "My Token Provider",
  "audience": [
    "28edee01-4d0d-46ed-b1ae-52139bc3b3ad"
  ],
  "userClaim": "preferred_username",
  "issuer": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/v2.0",
  "jwks": "https://login.microsoftonline.com/959d4644-91e6-4652-9d16-bddeb046c807/discovery/v2.0/keys",
  "type": "JWT",
  "state": "ENABLED"
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Update an External Token Provider's State

Update the state for the specified external token provider.

Method and URL

```
PATCH /api/v3/external-token-providers/{id}/state
```

Parameters

id Path String (UUID)

Unique identifier of the external token provider.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

state Body String

Current state of the external token provider: `ENABLED` or `DISABLED`.

Example: DISABLED

Example Request

```
curl -X PUT 'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa/state' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "state": "DISABLED"
}'
```

A successful request to update the state for an external token provider returns an empty response with the HTTP `204 No Content` status response code.

Response Status Codes

204 No Content

400 Bad Request

403 Forbidden

404 Not Found

Delete an External Token Provider

Delete the specified external token provider.

Method and URL

```
DELETE /api/v3/external-token-providers/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the external token provider that you want to delete.

Example: a32191f2-ed6-4533-9a17-1532eea015aa

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/external-token-providers/a32191f2-ed6-4533-9a17-1532eea015aa' \
--header 'Authorization: Bearer <dremioAccessToken>' \
```



```
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

403 Forbidden

Was this page helpful?

[\[Previous](#)

[Configuration](#)

[Options\]\(/current/reference/api/engine-management/engine-configuration-options\)\[Next](#)

[Job\]\(/current/reference/api/job/\)](#)

[External Token Providers Attributes](#)

[List All External Token Providers](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Create an External Token Provider](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve an External Token Provider by ID](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Update an External Token Provider](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

-

Update an External Token Provider's State

Parameters

Example Request

Response Status Codes

Delete an External Token Provider

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/job/>

Version: current [26.x]

On this page

Job

Use the Job API to get information about a specific job and cancel a running job.

note

To retrieve results for a specific job, use the [Job Results](#) endpoint.

Job Object

```
{
  "jobState": "COMPLETED",
  "rowCount": 1003904,
  "errorMessage": "",
  "startedAt": "2022-12-13T18:34:14.069Z",
  "endedAt": "2022-12-13T18:35:09.963Z",
  "acceleration": {
    "reflectionRelationships": [
      {
        "datasetId": "ef99ab32-aa47-4f4c-4d1c-d40f8035b846",
        "reflectionId": "63fd1c83-5cde-4133-9e2d-60543550580a",
        "relationship": "CONSIDERED"
      },
      {
        "datasetId": "596c489c-7949-485b-92a9-c32a4cb51fa2",
        "reflectionId": "65747723-2319-430p-8a36-3d40b26f45ae",
        "relationship": "MATCHED"
      }
    ]
  },
  "queryType": "UI_RUN",
```

```
"queueName": "LARGE",  
"queueId": "f6a5ef4f-ce5c-4be4-95b2-092c36721dc5",  
"resourceSchedulingStartedAt": "2022-12-13T18:34:14.977Z",  
"resourceSchedulingEndedAt": "2022-12-13T18:34:14.995Z",  
"cancellationReason": ""  
}
```

Job Attributes

jobState String

The job's status. Values `COMPLETED`, `CANCELED`, and `FAILED` are final; other values are considered in running state.

Enum: NOT_SUBMITTED, STARTING, RUNNING, COMPLETED, CANCELED, FAILED, CANCELLATION_REQUESTED, PLANNING, PENDING, METADATA_RETRIEVAL, QUEUED, ENGINE_START, EXECUTION_PLANNING, INVALID_STATE

Example: COMPLETED

rowCount Integer

For jobs with `COMPLETED` jobState, the number of rows the job returned. If jobState is not `COMPLETED`, rowCount value is `0`.

Example: 11

errorMessage String

For jobs with `FAILED` jobState, the error that caused the failure. For all other jobs, the errorMessage value is empty.

Example: Column 'user_id' not found in any table.

startedAt String

Date and time when the job started, in UTC format.

Example: 2022-12-09T20:16:15.694Z

endedAt String

Date and time when the job ended, in UTC format.

Example: 2022-12-09T20:16:19.939Z

acceleration Object

For jobs with applicable Reflections, provides more information about the Reflections and their relationships to the job. For jobs that do not have applicable Reflections, the response does not include the acceleration object.

queryType String

Job type. If the job's queryType is not set, the value is `UNKNOWN`.

Enum: UI_RUN, UI_PREVIEW, UI_INTERNAL_PREVIEW, UI_INTERNAL_RUN, UI_EXPORT, ODBC, JDBC, REST, ACCELERATOR_CREATE, ACCELERATOR_DROP, UNKNOWN, PREPARE_INTERNAL, ACCELERATOR_EXPLAIN, UI_INITIAL_PREVIEW

Example: UI_RUN

queueName String

Name of the workload management (WLM) queue to which the job was routed.

Example: SMALL

queueId String

ID of the workload management (WLM) queue to which the job was routed.

Example: f6a5ef4f-ce5c-4be4-95b2-092c36721dc5

resourceSchedulingStartedAt String

Date and time when the Dremio engine started scheduling the job.

Example: 2022-12-09T20:16:16.141Z

resourceSchedulingEndedAt String

Date and time when Dremio engine scheduling ended for the job.

Example: 2022-12-09T20:16:16.162Z

cancellationReason String

For canceled jobs, the reason for the cancellation. For all other jobs, the cancellationReason value is empty.

Example: Query was cancelled due to low memory.

Attributes of the `acceleration` Object

reflectionRelationships Array of Object

Information about the dataset, Reflection, and type of relationship for each applicable Reflection.

Attributes of Objects in the `reflectionRelationships` Array

datasetId String (UUID)

Unique identifier of the dataset associated with the Reflection.

Example: 596c489c-7949-485b-92a9-c32a4cb51fa2

reflectionId String (UUID)

Unique identifier of the Reflection.

Example: 65747723-2319-430p-8a36-3d40b26f45ae

relationship String

Type of relationship between the Reflection and the job.

Enum: CONSIDERED, MATCHED, CHOSEN

Example: MATCHED

Retrieve a Job

Retrieve the specified job.

Method and URL

```
GET /api/v3/job/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the job to retrieve. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

Example Request

```
curl -X GET 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response for a COMPLETED Job

```
{
  "jobState": "COMPLETED",
  "rowCount": 1003904,
  "errorMessage": "",
  "startedAt": "2022-12-13T18:34:14.069Z",
  "endedAt": "2022-12-13T18:35:09.963Z",
  "acceleration": {
    "reflectionRelationships": [
      {
        "datasetId": "ef99ab32-aa47-4f4c-4d1c-d40f8035b846",
        "reflectionId": "63fd1c83-2319-5962-8a36-60543550580a",
        "relationship": "CONSIDERED"
      },
      {
        "datasetId": "596c489c-7949-485b-92a9-c32a4cb51fa2",
        "reflectionId": "65747723-4133-9e2d-3k86-3d40b26f45ae",
        "relationship": "MATCHED"
      }
    ]
  }
}
```

```

    ]
  },
  "queryType": "UI_RUN",
  "queueName": "LARGE",
  "queueId": "f6a5ef4f-ce5c-4be4-95b2-092c36721dc5",
  "resourceSchedulingStartedAt": "2022-12-13T18:34:14.977Z",
  "resourceSchedulingEndedAt": "2022-12-13T18:34:14.995Z",
  "cancellationReason": ""
}

```

Example Response for a CANCELED Job

```

{
  "jobState": "CANCELED",
  "rowCount": 0,
  "errorMessage": "",
  "startedAt": "2023-02-01T15:07:16.165Z",
  "endedAt": "2023-02-01T15:07:18.691Z",
  "queryType": "UI_RUN",
  "queueName": "LARGE",
  "queueId": "6ed7841e-e446-4536-8d47-361508e78c18",
  "resourceSchedulingStartedAt": "2023-02-01T15:07:17.124Z",
  "resourceSchedulingEndedAt": "2023-02-01T15:07:17.140Z",
  "cancellationReason": "Query cancelled by user 'USERNAME'"
}

```

Example Response for a FAILED Job

```

{
  "jobState": "FAILED",
  "rowCount": 0,
  "errorMessage": "ExecutionSetupException: One or more nodes lost connectivity during query. Identified nodes were [automaster-2.c.dremio-1093.internal:0].",
  "startedAt": "2023-02-01T16:36:35.897Z",
  "endedAt": "2023-02-01T16:37:36.098Z",
  "queryType": "UI_RUN",
  "queueName": "LARGE",
  "queueId": "3d04235f-3610-4dd3-95b6-6a29542eb600",
  "resourceSchedulingStartedAt": "2023-02-01T16:36:37.389Z",
  "resourceSchedulingEndedAt": "2023-02-01T16:36:37.437Z",
  "cancellationReason": ""
}

```

Example Response for a RUNNING Job

```

{
  "jobState": "RUNNING",
  "rowCount": 2682474,
  "errorMessage": "",
  "startedAt": "2023-02-01T21:30:10.755Z",
  "queryType": "ACCELERATOR_CREATE",
  "queueName": "LARGE",

```

```
"queueId": "f64ff0a0-a925-4dc9-be60-e0703ce3aa24",  
"resourceSchedulingStartedAt": "2023-02-01T21:30:11.743Z",  
"resourceSchedulingEndedAt": "2023-02-01T21:30:11.798Z",  
"cancellationReason": ""  
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Cancel a Running Job

Cancel the specified running job.

note

Canceling a job does not delete the job object. You can still retrieve job objects for canceled jobs.

Method and URL

```
POST /api/v3/job/{id}/cancel
```

Parameters

id Path String (UUID)

Unique identifier of the job to cancel. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

Example request

```
curl -X POST 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00/cancel'  
\  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Example response

No response

Response Status Codes

204 No Content

400 Bad Request

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

External Token Providers](/current/reference/api/external-token-providers/)[Next

Job Results](/current/reference/api/job/job-results)

Job Attributes

Retrieve a Job

Parameters

Example Request

Response Status Codes

Cancel a Running Job

Parameters

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/ldap-authorization>

Version: current [26.x]

On this page

LDAP User Cache

Use the LDAP User Cache API to invalidate the authorization for all users and groups or a single user or group in the LDAP user cache.

note

You must be a member of the Dremio ADMIN role to send requests to the LDAP User Cache API.

Invalidate LDAP Authorization for All Users and Groups

Invalidate all users' and groups' LDAP authorizations.

Method and URL

```
DELETE /api/v3/cache/authorization
```

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/cache/authorization' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Invalidate LDAP Authorization for a Single User or Group by ID

Invalidate LDAP authorization for a single user or group by specifying the ID for the user or group.

note

If you do not provide the ID of a user or group in the request URL, Dremio invalidates the LDAP authorization for all users and groups in the cache.

Method and URL

```
DELETE /api/v3/cache/authorization/{id}
```

Parameters

name Path String (UUID)

Unique identifier of the Dremio user or group whose LDAP authorization you want to invalidate.

Example: 2k8bdk96-b267-4d56-9154-e48v5884h5i8

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/cache/authorization/2k8bdk96-b267-4d56-9154-e48v5884h5i8' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Invalidate LDAP Authorization for a Single User or Group by Name

Invalidate LDAP authorization for a single user or group by specifying the name for the user or group.

note

If you do not provide the name of a user or group in the request URL, Dremio invalidates the LDAP authorization for all users and groups in the cache.

Method and URL

```
DELETE /api/v3/cache/authorization/{name}
```

Parameters

name Path String

Name for the Dremio user or group whose LDAP authorization you want to invalidate.

Example: exampleuser1

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/cache/authorization/exampleuser1' \
--header 'Authorization: Bearer <dremioAccessToken>' \
```

```
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

Job]([current/reference/api/job/])[Next

Node Collections]([current/reference/api/nodeCollections/)

Invalidate LDAP Authorization for All Users and Groups

Example

Response Status Codes

Invalidate LDAP Authorization for a Single User or Group by ID

Parameters

Example

Response Status Codes

Invalidate LDAP Authorization for a Single User or Group by Name

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/nodeCollections/>

Version: current [26.x]

On this page

Node Collections

Use the Node Collections API to deny access to Dremio cluster nodes and retrieve the currently denied nodes for the Dremio instance.

The Node Collections API does not have a corresponding Node Collection object. The endpoints return a list of the currently denied nodes in the response.

note

You must be a member of the Dremio ADMIN role to send requests to the Node Collections API.

Deny Nodes

Deny access to the specified Dremio cluster nodes.

Method and URL

```
POST /api/v3/nodeCollections/blacklist
```

The request body is a comma-separated list of the names for the nodes that you want to deny, including any currently denied nodes that should remain denied. Format the request body as an array of string, with each node name in double quotes. Use a comma to separate each node name in the list.

Any nodes omitted from the request body, including currently denied nodes, will be allowed.

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/nodeCollections/blacklist' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '["localhost-1.c.dremio-1093.external",
"localhost-2.c.dremio-1093.external"]'
```

Response

```
[
  "localhost-1.c.dremio-1093.external",
  "localhost-2.c.dremio-1093.external"
]
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Retrieve Denied Nodes

Invalidate the LDAP authorization for a specific user or group by ID.

Method and URL

```
GET /api/v3/nodeCollections/blacklist
```

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/nodeCollections/blacklist' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
[  
  "localhost-1.c.dremio-1093.external",  
  "localhost-2.c.dremio-1093.external"  
]
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Allow All Nodes

Allow access to all Dremio cluster nodes.

Method and URL

```
POST /api/v3/nodeCollections/blacklist
```

To allow all nodes, send an empty array in the request body.

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/nodeCollections/blacklist' \  
'[]'
```

```
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json' \  
--data-raw '[]'
```

Response

```
[]
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

LDAP User Cache](/current/reference/api/ldap-authorization)[Next

OAuth Token](/current/reference/api/oauth-token)

Deny Nodes

Example

Response Status Codes

Retrieve Denied Nodes

Example

Response Status Codes

Allow All Nodes

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/oauth-token>

Version: current [26.x]

On this page

OAuth Token Enterprise

Clients use the `/oauth/token` endpoint to request OAuth access tokens for creating connections to Dremio.

The Dremio OAuth authorization server is located at `{hostname}/oauth/token`. After a valid request, Dremio returns an OAuth access token in a JSON object, along with the token lifetime and other metadata.

OAuth Token Object

```
{
  "access_token": "eyJz93a...k4laUWw",
  "expires_in": 3599,
  "token_type": "Bearer",
  "issued_token_type": "urn:ietf:params:oauth:token-type:access_token",
  "scope": "dremio.all offline_access",
  "refresh_token": "ism9f1nf68lre2salj0tq0btor"
}
```

Object Attributes

`access_token` String

The returned access token. A client application passes this token when connecting with Dremio.

Example: `eyJz93a...k4laUWw`

`expires_in` Integer

The access token lifetime in seconds. The default lifetime is 3600 seconds (1 hour).

Example: `3599`

`token_type` String

`Bearer` for all access tokens.

`issued_token_type` String

`urn:ietf:params:oauth:token-type:access_token` for access tokens.

`scope` String

A space-separated list of case-sensitive scopes associated with the OAuth access token. If the value contains multiple space-delimited strings, their order does not matter, and each scope adds an access range to the requested scope.

Example: `dremio.all offline_access`

`refresh_token` String Optional

An optional OAuth refresh token is a credential for obtaining a new access token when the current one expires. The refresh token field is included when the request `scope` includes `offline_access`.

Example: `ism9f1nf68lre2salj0tq0btor`

Obtain Tokens via Client ID and Client Secret

The Client ID and Client Secret are used to obtain a short-lived access token that provides secure, time-limited access to resources.

See [User Management](#) to create service users for machine-to-machine applications.

Method and URL

```
POST /{hostname}/oauth/token
```

Parameters

client_id Body String

The unique identifier for your [service user](#), assigned by Dremio.

Example: `d3821306-1f75-4233-9c42-1a295f219850`

client_secret Body String

The confidential credential that proves your application's identity. This secret must be kept secure and should never be exposed in client-side code or version control.

Example:

`gkuzJELITySLGY7IJYRdt6EVW1YAJf7BLKROnH81mUDBq7rmVhT6OZU0G+kf5g==`

grant_type Body String

Valid value is `client_credentials`.

scope Body String

Defines the level of access being requested. The scope must be set to `dremio.all`, which grants full access to Dremio resources.

Example

cURL

Python

Request with Client Credentials

```
#!/bin/bash

# Get OAuth token
curl -s -X POST "https://$DREMIO_HOST:9047/oauth/token" \
  -H "Content-Type: application/x-www-form-urlencoded" \
  --data-urlencode "grant_type=client_credentials" \
  --data-urlencode "client_id=$CLIENT_ID" \
  --data-urlencode "client_secret=$CLIENT_SECRET" \
  --data-urlencode "scope=dremio.all"
```


Response

```
{
  "access_token": "eyJz93a...k4laUWw",
  "expires_in": 3599,
  "token_type": "Bearer",
  "issued_token_type": "urn:ietf:params:oauth:token-type:access_token",
  "scope": "dremio.all offline_access",
  "refresh_token": "ism9f1nf68lre2salj0tq0btor"
}
```

Request with Client Credentials

```
import requests
import json

class DremioClient:
    def __init__(self, config_file='config.json'):
        self.config = self._load_config(config_file)
        self.client_id = self.config['client_id']
        self.client_secret = self.config['client_secret']
        self.dremio_host = self.config['dremio_host']

    def _load_config(self, config_file):
        try:
            with open(config_file, 'r') as f:
                return json.load(f)
        except FileNotFoundError:
            raise FileNotFoundError(f"Config file '{config_file}' not found")
        except json.JSONDecodeError:
            raise ValueError(f"Invalid JSON in config file '{config_file}'")

    def get_token(self):
        token_url = f"https://{self.dremio_host}:9047/oauth/token"
        data = {
            'grant_type': 'client_credentials',
            'client_id': self.client_id,
            'client_secret': self.client_secret,
            'scope': 'dremio.all'
        }

        response = requests.post(token_url, data=data)
        response.raise_for_status()
        return response.json()['access_token']

# Usage
client = DremioClient('dremio-config.json')
token = client.get_token()
```

The lifetime of this token is one hour.

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
403 Forbidden

Obtain Tokens via Username and Password

For users authenticated locally or with an enterprise LDAP server, Dremio can provide OAuth access tokens using your username and password.

Method and URL

```
POST /{hostname}/oauth/token
```

Parameters

username Body String

The username that will be embedded in the token.

Example: `dremio_user`

password Body String

The user's password.

Example: `dremio123`

grant_type Body String

The type of authentication method. For this method, the grant_type is `password`.

scope Body String

Must contain `dremio.all`. If the scope also contains `offline_access`, Dremio will return a refresh token with the access token.

Example: `dremio.all offline_access`

Example

curl

Python

Request Using User Credentials

```
curl -X POST 'https://{hostname}/oauth/token' \  
--header 'Content-Type: application/x-www-form-urlencoded' \  
--data-urlencode 'username=dremio' \  
--data-urlencode 'password=dremio123' \  

```

```
--data-urlencode 'grant_type=password' \  
--data-urlencode 'scope=dremio.all offline_access'
```

Response

```
{  
  "access_token": "eyJz93a...k4laUWw",  
  "expires_in": 3599,  
  "token_type": "Bearer",  
  "issued_token_type": "urn:ietf:params:oauth:token-type:access_token",  
  "scope": "dremio.all offline_access",  
  "refresh_token": "ism9f1nf68lre2salj0tq0btor"  
}
```

Request Using User Credentials

```
import requests  
  
form_data = {  
    "username": "dremio_user",  
    "password": "dremio123",  
    "grant_type": "password",  
    "scope": "dremio.all offline_access"  
}  
  
request_header = {"Content-Type": "application/x-www-form-urlencoded"}  
  
token_exchange_response = requests.post(  
    "https://{hostname}/oauth/token", headers=request_header, data=form_data)  
  
if token_exchange_response.status_code == 200:  
    token_exchange_response_json = token_exchange_response.json()  
    dremio_access_token = token_exchange_response_json["access_token"]  
    dremio_access_token_expires_in_sec = token_exchange_response_json["expires_in"]  
    dremio_refresh_token = token_exchange_response_json["refresh_token"]  
else:  
    print("Error: " + str(token_exchange_response.status_code))
```

The lifetime of this token is one hour.

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

500 Internal Server Error

Exchange a Refresh Token

If a refresh token was requested with the OAuth access token using the `offline_access` scope, the refresh token can be exchanged for fresh access tokens until the refresh token expires. Each refresh token has a lifetime of 30 days.

Method and URL

```
POST /{hostname}/oauth/token
```

Parameters

grant_type Body String

The type of token being exchanged. For a token refresh, the grant_type is `refresh_token`.

client_id Body String

The username from the original token request.

Example: `dremio_user`

refresh_token Body String

The refresh token returned with the original token request.

Example: `ism9f1nf68lre2salj0tq0btor`

Example

cURL

Python

Request Using a Refresh Token

```
curl -X POST 'https://{hostname}/oauth/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'grant_type=refresh_token' \
--data-urlencode 'client_id=dremio' \
--data-urlencode 'refresh_token=ism9f1nf68lre2salj0tq0btor'
```

Response

```
{
  "access_token": "eyJz93a...k4laUWw",
  "expires_in": 3599,
  "token_type": "Bearer",
  "issued_token_type": "urn:ietf:params:oauth:token-type:access_token",
  "scope": "dremio.all offline_access"
}
```

Request Using a Refresh Token

```
import requests

form_data = {
    "grant_type": "refresh_token",
    "client_id": "dremio",
    "refresh_token": "ism9f1nf68lre2salj0tq0btor"
}

request_header = {"Content-Type": "application/x-www-form-urlencoded"}

token_exchange_response = requests.post(
    "https://{hostname}/oauth/token", headers=request_header, data=form_data)

if token_exchange_response.status_code == 200:
    token_exchange_response_json = token_exchange_response.json()
    dremio_access_token = token_exchange_response_json["access_token"]
    dremio_access_token_expires_in_sec = token_exchange_response_json["expires_in"]
else:
    print("Error: " + str(token_exchange_response.status_code))
```

The lifetime of this token is one hour.

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

500 Internal Server Error

Exchange an External JWT

Clients who authenticate with an OIDC external token provider can exchange their JWT for an OAuth access token that can be used to create connections to Dremio.

Method and URL

```
POST /{hostname}/oauth/token
```

Parameters

subject_token Body String

The external JWT obtained from an OIDC provider like Microsoft Entra ID.

Example: `eyJ0eXAiOiJKV1QiLCJhbGciOiJSUz...DYynR2lK6xB8xrAprgPA`

subject_token_type Body String

The type of subject token used. For an external JWT, the subject_token_type is `urn:ietf:params:oauth:token-type:jwt`.

grant_type Body String

The type being granted. For a token exchange, the grant_type is `urn:ietf:params:oauth:grant-type:token-exchange`.

scope Body String

The scope of the request. For a token exchange, the scope is `dremio.all`.

Example

CURL

Python

Request Using an External JWT

```
curl -X POST 'https://{hostname}/oauth/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'subject_token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUz...DYynR2lK6xB8xrAprgPA' \

--data-urlencode 'subject_token_type=urn:ietf:params:oauth:token-type:jwt' \
--data-urlencode 'grant_type=urn:ietf:params:oauth:grant-type:token-exchange' \
--data-urlencode 'scope=dremio.all'
```

Response

```
{
  "access_token": "eyJz93a...k4laUWw",
  "expires_in": 3599,
  "token_type": "Bearer",
  "issued_token_type": "urn:ietf:params:oauth:token-type:access_token",
  "scope": "dremio.all"
}
```

Request Using an External JWT

```
import requests

form_data = {
    "subject_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUz...DYynR2lK6xB8xrAprgPA",
    "subject_token_type": "urn:ietf:params:oauth:token-type:jwt",
    "grant_type": "urn:ietf:params:oauth:grant-type:token-exchange",
    "scope": "dremio.all"
}

request_header = {"Content-Type": "application/x-www-form-urlencoded"}
```

```
token_exchange_response = requests.post(
    "https://{hostname}/oauth/token", headers=request_header, data=form_data)

if token_exchange_response.status_code == 200:
    token_exchange_response_json = token_exchange_response.json()
    dremio_access_token = token_exchange_response_json["access_token"]
    dremio_access_token_expires_in = token_exchange_response_json["expires_in"]
else:
    print("Error: " + str(token_exchange_response.status_code))
```

The lifetime of this token is the time remaining on the external JWT, up to one hour.

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

500 Internal Server Error

Exchange a PAT

Exchanging a personal access token (PAT) for an OAuth access token provides the security benefit of a shorter token lifetime while allowing a client application to access protected resources with a more controlled access mechanism. OAuth access tokens also perform better due to faster validation time than PATs.

Method and URL

POST /{hostname}/oauth/token

Parameters

subject_token Body String

The personal access token to be exchanged.

Example: `wPTsz2YrTVWQ7fw436Ec...911rJzUm6Xs1XrvU+w==`

subject_token_type Body String

The type of subject token used. For a PAT, the subject_token_type is `urn:ietf:params:oauth:token-type:dremio:personal-access-token`.

grant_type Body String

The type being granted. For a token exchange, the `grant_type` is `urn:ietf:params:oauth:grant-type:token-exchange`.

`scope` Body String

The scope of the request. For a token exchange, the scope is `dremio.all`.

Example

curl

Python

Request Using a PAT

```
curl -X POST 'https://{hostname}/oauth/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode
'subject_token=wPTsz2YrTVWQ7fw436Ec7Vs16TAWeRyojniYNXED1THt911rJzUm6Xs1XrvU+w==' \
--data-urlencode
'subject_token_type=urn:ietf:params:oauth:token-type:dremio:personal-access-token' \
--data-urlencode 'grant_type=urn:ietf:params:oauth:grant-type:token-exchange' \
--data-urlencode 'scope=dremio.all'
```

Response

```
{
  "access_token": "eyJz93a...k4laUWw",
  "expires_in": 3599,
  "token_type": "Bearer",
  "issued_token_type": "urn:ietf:params:oauth:token-type:access_token",
  "scope": "dremio.all"
}
```

Request Using a PAT

```
import requests

form_data = {
    "subject_token": "wPTsz2YrTVWQ7fw436Ec...911rJzUm6Xs1XrvU+w==",
    "subject_token_type":
"urn:ietf:params:oauth:token-type:dremio:personal-access-token",
    "grant_type": "urn:ietf:params:oauth:grant-type:token-exchange",
    "scope": "dremio.all"
}

request_header = {"Content-Type": "application/x-www-form-urlencoded"}

token_exchange_response = requests.post(
    "https://{hostname}/oauth/token", headers=request_header, data=form_data)

if token_exchange_response.status_code == 200:
    token_exchange_response_json = token_exchange_response.json()
```



```
dremio_access_token = token_exchange_response_json["access_token"]
dremio_access_token_expires_in_sec = token_exchange_response_json["expires_in"]
else:
    print("Error: " + str(token_exchange_response.status_code))
```

The lifetime of this token is the remaining lifetime of the PAT used in the exchange, up to one hour.

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 500 Internal Server Error

Exchange a PAT with User Impersonation

Dremio's [inbound user impersonation](#) feature allows a privileged user, called a proxy user, to run queries on Dremio as a second target user using the target user's privileges. The privileged user obtains the necessary privileges to impersonate a second target user from an inbound impersonation policy created by the Dremio administrator. Once the inbound impersonation policy is in place, the proxy user runs queries as the target user as allowed by the target user's privileges.

Inbound Impersonation Policy

```
ALTER SYSTEM SET "exec.impersonation.inbound_policies"='[
  {
    proxy_principals:{
      users:["mark"]
    },
    target_principals:{
      users:["sharedaccessuser"]
    }
  }
]'
```

Method and URL

```
POST /{hostname}/oauth/token
```

Parameters

subject_token Body String

The target principal's username.

Example: `sharedaccessuser`

subject_token_type Body String

The type of subject token used. For a target principal, the subject_token_type is `urn:ietf:params:oauth:token-type:dremio:subject`.

actor_token Body String

The proxy principal's personal access token.

Example: `S0ilZcxyQCeH6m8hoNeCdVYwc...BjsxFsJfvCVhHsQ==`

actor_token_type Body String

The type of actor token. For a PAT, the actor_token_type is `urn:ietf:params:oauth:token-type:dremio:personal-access-token`.

grant_type Body String

The type being granted. For a token exchange, the grant_type is `urn:ietf:params:oauth:grant-type:token-exchange`.

scope Body String

The scope of the request. For a token exchange, the scope is `dremio.all`.

Example

curl

Python

Request Using a PAT with Impersonation

```
curl -X POST 'http://localhost:9047/oauth/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--header 'Accept: application/json' \
--data-urlencode 'subject_token=sharedaccessuser' \
--data-urlencode 'subject_token_type=urn:ietf:params:oauth:token-type:dremio:subject' \
--data-urlencode
'actor_token=wPTsz2YrTVWQ7fw436Ec7Vs16TAWeRyojniYNXED1THt911rJzUm6Xs1XrvU+w==' \
--data-urlencode
'actor_token_type=urn:ietf:params:oauth:token-type:dremio:personal-access-token' \
--data-urlencode 'grant_type=urn:ietf:params:oauth:grant-type:token-exchange' \
--data-urlencode 'scope=dremio.all'
```

Response

```
{
  "access_token": "eyJz93a...k4laUWw",
  "expires_in": 3599,
  "token_type": "Bearer",
```

```
"issued_token_type": "urn:ietf:params:oauth:token-type:access_token",
"scope": "dremio.all"
}
```

Request Using a PAT with Impersonation

```
import requests

form_data = {
    "subject_token": "sharedaccessuser",
    "subject_token_type": "urn:ietf:params:oauth:token-type:dremio:subject",
    "actor_token": "wPTsz2YrTVWQ7fw436Ec7Vs16TAWeRyojniYNXED1THt911rJzUm6Xs1XrvU+w==",
    "actor_token_type": "urn:ietf:params:oauth:token-type:dremio:personal-access-token",

    "grant_type": "urn:ietf:params:oauth:grant-type:token-exchange",
    "scope": "dremio.all"
}

request_header = {"Content-Type": "application/x-www-form-urlencoded"}

token_exchange_response = requests.post(
    "https://{hostname}/oauth/token", headers=request_header, data=form_data)

if token_exchange_response.status_code == 200:
    token_exchange_response_json = token_exchange_response.json()
    dremio_access_token = token_exchange_response_json["access_token"]
    dremio_access_token_expires_in_sec = token_exchange_response_json["expires_in"]
else:
    print("Error: " + str(token_exchange_response.status_code))
```

The lifetime of this token is the remaining lifetime of the PAT used in the exchange, up to one hour.

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

500 Internal Server Error

Was this page helpful?

[\[Previous](#)

[Node Collections\]\(/current/reference/api/nodeCollections/\)](#)[\[Next](#)

[Personal Access Token\]\(/current/reference/api/personal-access-token\)](#)

Object Attributes

Obtain Tokens via Client ID and Client Secret

Parameters

Example

Response Status Codes

Obtain Tokens via Username and Password

Parameters

Example

Response Status Codes

Exchange a Refresh Token

Parameters

Example

Response Status Codes

Exchange an External JWT

Parameters

Example

Response Status Codes

Exchange a PAT

Parameters

Example

Response Status Codes

Exchange a PAT with User Impersonation

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/personal-access-token>

Version: current [26.x]

On this page

Personal Access Token Enterprise

Use the API to create and retrieve personal access tokens (PATs) for the current Dremio user and delete PATs for any Dremio user.

You must enable the use of PATs prior to using these requests.

Personal Access Token Object

```
{
  "tid": "98ec8f42-7764-4d9d-af5a-693f1f1cc444",
  "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "label": "Tableau",
  "createdAt": "2023-02-19T15:41:15.323Z",
  "expiresAt": "2023-03-21T15:41:15.323Z"
}
```

Attributes

tid String (UUID)

Unique identifier of the PAT.

Example: 98ec8f42-7764-4d9d-af5a-693f1f1cc444

uid String (UUID)

Unique identifier of the user.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

label String

User-provided name of the PAT.

Example: Tableau

createdAt String

Date and time that the PAT was created, in UTC format.

Example: 2023-02-19T15:41:15.323Z

expiresAt String

Date and time that the PAT will expire, in UTC format.

Example: 2023-03-21T15:41:15.323Z

Listing All PATs for One User

List all PATs for the specified user. Users can retrieve only their own PAT metadata. The `ADMIN` role is required to retrieve PAT metadata owned by other users.

Method and URL

```
GET /api/v3/user/{user-id}/token
```

Parameters

id Path String (UUID)

Dremio identifier of the named user.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "tid": "98ec8f42-7764-4d9d-af5a-693f1f1cc444",
      "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
      "label": "Tableau",
      "createdAt": "2023-02-19T15:41:15.323Z",
      "expiresAt": "2023-03-21T15:41:15.323Z"
    },
    {
      "tid": "3b76a1e4-6539-46de-8f06-b7c41c71b61e",
      "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
      "label": "Test Nessie Source",
      "createdAt": "2023-03-02T19:39:52.159Z",
      "expiresAt": "2023-04-01T19:39:52.159Z"
    },
    {
      "tid": "9376ef58-7b4c-2419-b1cb-a4ce4c53dfa7",
      "uid": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
      "label": "Feature Testing",
      "createdAt": "2023-03-07T14:47:08.211Z",
      "expiresAt": "2023-09-03T14:47:08.211Z"
    }
  ]
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Create a PAT

Create a personal access token. Users can create only their own PATs and only use the PATs they create. Administrators cannot create PATs for other users or distribute PATs to other users.

Method and URL

```
POST /api/v3/user/{user-id}/token
```

Parameters

user-id Path String (UUID)

Your Dremio user identifier.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

label Body String

User-provided label for the resulting PAT.

Example: Feature Testing

millisecondsToExpire Body String

The number of milliseconds until the PAT expires. The value `15552000000` is 180 days.

Example: 15552000000

Example Request to Create a PAT

```
curl -X POST 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "label": "Feature Testing",
  "millisecondsToExpire": 15552000000
}'
```

Example Response

```
EXAMPLETOKEN7TjB3mfPS6AZQ5aPcXPmJS2ofXpLL86dmpDXRbKKi52BQdthnk==
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Deleting a Single PAT

Delete a single PAT for the specified user. Users can delete only their own PATs. The `ADMIN` role is required to delete PATs owned by other users.

Method and URL

```
DELETE /api/v3/user/{user-id}/token/{token-id}
```

Parameters

user-id Path String (UUID)

Unique identifier of the user whose PAT will be deleted.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

token-id Path String (UUID)

Unique identifier of the PAT you want to delete.

Example: 98ec8f42-7764-4d9d-af5a-693f1f1cc444

Example Request

```
curl -X DELETE \
'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token/98ec8f42-7764-4d9d-af5a-693f1f1cc444' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Dremio does not provide a response to this command.

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Delete All PATs for One User

Delete all PATs for the specified user. Users can delete only their own PATs. The `ADMIN` role is required to delete PATs owned by other users.

Method and URL

```
DELETE /api/v3/user/{user-id}/token
```

Parameters

user-id Path String (UUID)

Unique identifier of the user whose PATs will be deleted.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example Request

```
curl -X DELETE \  
'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/token' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Dremio does not provide a response to this command.

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Delete All PATs

Delete all PATs in the system for all users, including users with the `ADMIN` role. The user invoking the command must be in the `ADMIN` role.

warning

This command deletes all PATs for all users in the system. This action cannot be undone.

Method and URL

```
DELETE /api/v3/token
```

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/token' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

OAuth Token](/current/reference/api/oauth-token)[Next

Reflection](/current/reference/api/reflections/)

Attributes

Listing All PATs for One User

Parameters

Example

Response Status Codes

Create a PAT

Parameters

Response Status Codes

Deleting a Single PAT

Parameters

Example Request

Response Status Codes

Delete All PATs for One User

Parameters

-

Example Request

Response Status Codes

Delete All PATs

Example Request

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/reflections/>

Version: current [26.x]

On this page

Reflection

Use the Reflection API to retrieve a list of raw and aggregation Reflections, retrieve individual Reflections, and create, update, and delete Reflections.

A Reflection is an optimized materialization of source data or a query, similar to a materialized view, that is derived from an existing table or view. The query optimizer can accelerate queries by using one or more Reflections to partially or entirely satisfy the queries rather than running queries against the raw data in the data source that underlies the table or view.

Reflection Object (Raw Reflection)

```
{
  "id": "7a380a24-3b63-436c-9ea0-63cb534cc404",
  "type": "RAW",
  "reflectionMode": "Manual",
  "name": "Raw Reflection",
  "tag": "ureIY76RT7Y=",
  "createdAt": "2023-01-30T14:11:43.826Z",
  "updatedAt": "2023-01-30T14:11:43.826Z",
  "datasetId": "tk973df7-ddf7-4d1e-fa9e-bccf28ae253f",
  "currentSizeBytes": 4393709246,
  "totalSizeBytes": 4393709246,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "failureCount": 0,
    "lastDataFetch": "2023-01-30T14:11:51.801Z",
    "expiresAt": "2023-01-30T17:11:51.801Z"
  },
  "displayFields": [
```

```

    {
      "name": "pickup_datetime"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "trip_distance_mi"
    },
    {
      "name": "fare_amount"
    },
    {
      "name": "tip_amount"
    },
    {
      "name": "total_amount"
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "partitionFields": [
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "pickup_datetime"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
}

```

Reflection Object (Aggregation Reflection)

```

{
  "id": "95dda9dd-2371-467f-b68d-fc4c5ea57a8b",
  "type": "AGGREGATION",
  "reflectionMode": "Autonomous",
  "name": "Aggregation Reflection",
  "tag": "ZpzGgxw2l04=",
  "createdAt": "2022-07-05T19:19:40.244Z",
  "updatedAt": "2023-01-10T17:12:40.244Z",
  "datasetId": "df99ab32-c2d4-4d1c-9e91-2c8be861bb8a",
  "currentSizeBytes": 18639885,
  "totalSizeBytes": 142639924,
}

```

```

"enabled": true,
"arrowCachingEnabled": false,
"status": {
  "config": "OK",
  "refresh": "SCHEDULED",
  "availability": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "failureCount": 0,
  "lastDataFetch": "2023-01-10T17:12:40.244Z",
  "expiresAt": "3022-07-05T19:19:40.244Z"
},
"dimensionFields": [
  {
    "name": "pickup_date"
  },
  {
    "name": "pickup_datetime",
    "granularity": "DATE"
  },
  {
    "name": "dropoff_date"
  },
  {
    "name": "dropoff_datetime",
    "granularity": "DATE"
  },
  {
    "name": "passenger_count"
  },
  {
    "name": "total_amount"
  },
  {
    "name": "trip_distance_mi"
  }
],
"measureFields": [
  {
    "name": "passenger_count",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "trip_distance_mi",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "fare_amount",
    "measureTypeList": [

```

```

        "SUM",
        "COUNT"
    ]
},
{
    "name": "surcharge",
    "measureTypeList": [
        "SUM",
        "COUNT"
    ]
},
{
    "name": "tip_amount",
    "measureTypeList": [
        "SUM",
        "COUNT"
    ]
},
{
    "name": "total_amount",
    "measureTypeList": [
        "SUM",
        "COUNT"
    ]
}
],
"distributionFields": [
    {
        "name": "trip_distance_mi"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "dropoff_date"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Reflection Attributes

id String (UUID)

Unique identifier of the Reflection.

Example: 95dda9dd-2371-467f-b68d-fc4c5ea57a8b

type String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

reflectionMode String

How the Reflection was created. Reflections can be [manually created](#) or [Autonomous](#)

example: "Manual", "Autonomous"

name String

User-provided name for the Reflection. For Reflections created in the Dremio UI, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: Aggregation Reflection

tag String

Unique identifier of the Reflection instance. Dremio changes the tag whenever the Reflection changes and uses the tag to ensure that PUT requests apply to the most recent version of the Reflection.

Example: ZpzGgxw2l04=

createdAt String

Date and time that the Reflection was created, in UTC format.

Example: 2022-07-05T19:19:40.244Z

updatedAt String

Date and time that the Reflection was last updated, in UTC format.

Example: 2023-01-10T17:12:40.244Z

datasetId String (UUID)

Unique identifier of the anchor dataset that is associated with the Reflection.

Example: df99ab32-c2d4-4d1c-9e91-2c8be861bb8a

currentSizeBytes Integer

Data size of the latest Reflection job (if one exists), in bytes.

Example: 18639885

totalSizeBytes Integer

Data size of all Reflection jobs that have not been pruned (if any exist), in bytes.

Example: 142639924

enabled Boolean

If the Reflection is available for accelerating queries, the value is `true`. Otherwise, the value is `false`.

Example: true

arrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, the value is `true`. Otherwise, the value is `false`.

Example: false

status Object

Information about the status of the Reflection.

Example: `{ "config": "OK", "refresh": "SCHEDULED", "availability": "AVAILABLE", "combinedStatus": "CAN_ACCELERATE", "failureCount": 0, "lastDataFetch": "2023-01-10T17:12:40.244Z", "expiresAt": "3022-07-05T19:19:40.244Z" }`

displayFields Array of Object

Information about the fields displayed from the anchor dataset. Each displayFields object contains one attribute: name. Valid only for raw Reflections.

Example: `[{ "name": "pickup_datetime", }, { "name": "passenger_count", }, { "name": "trip_distance_mi", }, { "name": "fare_amount", }, { "name": "tip_amount", }, { "name": "total_amount", }]`

dimensionFields Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each dimensionFields object contains two attributes: name and granularity. Valid only for aggregation Reflections.

Example: `[{ "name": "pickup_date", "granularity": "DATE", }, { "name": "pickup_datetime", "granularity": "DATE", }, { "name": "dropoff_date", "granularity": "DATE", }, { "name": "dropoff_datetime", "granularity": "DATE", }, { "name": "passenger_count", "granularity": "DATE", }, { "name": "total_amount", "granularity": "DATE", }]`

measureFields Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each measureFields object contains two attributes: name and measureTypeList.

Valid only for aggregation Reflections.

```
Example: [{"name": "passenger\_count","measureTypeList": ["SUM",
"COUNT"]}, {"name": "trip\_distance\_mi","measureTypeList": ["SUM",
"COUNT"]}, {"name": "fare\_amount","measureTypeList": ["SUM", "COUNT"]}, {"name":
"surcharge","measureTypeList": ["SUM", "COUNT"]}, {"name":
"tip\_amount","measureTypeList": ["SUM", "COUNT"]}, {"name":
"total\_amount","measureTypeList": ["SUM", "COUNT"]}]
```

distributionFields Array of Object

Information about the distribution fields from the anchor dataset used in the Reflection. Distribution fields allow data from multiple datasets to be co-located and co-partitioned across nodes to minimize data movement during join operations. Each distributionFields object contains one attribute: name.

```
Example: [{"name": "trip\_distance\_mi"}, {"name": "total\_amount"}]
```

partitionFields Array of Object

Information about the fields from the anchor dataset used to partition data in the Reflection. Each field name is listed as an individual object. For more information, read [Horizontally Partition Reflections that Have Many Rows](#).

```
Example: [{"name": "dropoff\_date"}, {"name": "passenger\_count"}]
```

sortFields Array of Object

Information about the fields from the anchor dataset used for sorting in the Reflection. Each sortFields object contains one attribute: name. For more information, read [Sort Reflections on High-Cardinality Fields](#).

```
Example: [{"name": "trip\_distance\_mi"}]
```

partitionDistributionStrategy String

Method used to optimize data compression when executing Reflections. `CONSOLIDATED` means Dremio minimizes the number of files produced. The query threads pool the data and ensure that the fewest number of files are written to the Reflection store. Optimizing for a smaller number of files generally improves read performance because users can perform fewer searches for a given query. `STRIPED` means Dremio minimizes the time required to refresh the Reflection. Each final-stage query thread opens its own writers to write the data, which can result in many small files if each query thread contains a small amount of data.

Enum: CONSOLIDATED, STRIPED

Example: CONSOLIDATED

canView Boolean

If you can view Reflections on all datasets of a source, system, space, or folder, the value is `true`. Otherwise, the value is `false`.

Example: true

canAlter Boolean

If you can create, edit, and view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

entityType String

Type of the object. For Reflection objects, the entityType is ``reflection``.

Example: reflection

Attributes of the ``status`` Object

config String

Status of the Reflection configuration. If the value is ``OK``, the Reflection configuration is free of errors. If the value is ``INVALID``, the Reflection configuration contains one or more errors.

Enum: OK, INVALID

Example: OK

refresh String

Status of the Reflection refresh.

``GIVEN_UP``: Dremio attempted to refresh the Reflection multiple times, but each attempt has failed and Dremio will not make further attempts.

``MANUAL``: Refresh period is set to 0, so you must use the Dremio UI to manually refresh the Reflection.

``RUNNING``: Dremio is currently refreshing the Reflection.

``SCHEDULED``: The Reflection refreshes according to a schedule.

Enum: GIVEN_UP, MANUAL, RUNNING, SCHEDULED

Example: SCHEDULED

availability String

Status of the Reflection's availability for accelerating queries.

Enum: NONE, EXPIRED, AVAILABLE

Example: AVAILABLE

combinedStatus String

Status of the Reflection based on a combination of config, refresh, and availability.

``CAN_ACCELERATE``: The Reflection is fully functional.

``CAN_ACCELERATE_WITH_FAILURES``: The most recent refresh failed to obtain a status, but Dremio still has a valid materialization.

``CANNOT_ACCELERATE_MANUAL``: The Reflection is unable to accelerate any queries, and the ``Never Refresh`` option is selected for the refresh policy.

``CANNOT_ACCELERATE_SCHEDULED``: The Reflection is currently unable to accelerate any queries, but it has been scheduled for a refresh at a future time.

``DISABLED``: The Reflection has been manually disabled.

``EXPIRED``: The Reflection has expired and cannot be used.

``FAILED``: The attempt to refresh the Reflection has failed, typically three times in a row. The Reflection is still usable.

``INVALID``: The Reflection is invalid because the underlying dataset has changed.

``REFRESHING``: The Reflection is currently being refreshed.

Example: `CAN_ACCELERATE`

failureCount Integer

Number of times that an attempt to refresh the Reflection failed.

Example: 0

lastDataFetch String

Date and time that the Reflection data was last refreshed, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetch value is ``1969-12-31T23:59:59.999Z``.

Example: 2023-01-10T17:12:40.244Z

expiresAt String

Date and time that the Reflection expires, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetch value is ``1969-12-31T23:59:59.999Z``.

Example: 3022-07-05T19:19:40.244Z

Attributes of Objects in the ``displayFields`` Array

name String

Name of the field from the anchor dataset that is displayed in the raw Reflection.

Example: `passenger_count`

Attributes of Objects in the ``dimensionFields`` Array

name String

Name of the field from the anchor dataset that is configured as a dimension for the Reflection.

Example: `pickup_date`

granularity String

Grouping used for the dimension field. When timestamp and date fields are configured as dimensions, Dremio can automatically extract and use the day-level date value (`DATE`) or use the field's original value (`NORMAL`).

Enum: DATE, NORMAL

Example: DATE

Attributes of Objects in the `measureFields` Array

name String

Name of the field from the anchor dataset that is configured as a measure for the Reflection.

Example: passenger_count

measureTypeList Array of String

Types of calculations for which Dremio uses the specified measure field.

Enum: APPROX_COUNT_DISTINCT, MIN, MAX, UNKNOWN, SUM, COUNT

Example: ["SUM","COUNT"]

Attributes of Objects in the `distributionFields` Array

name String

Name of the field from the anchor dataset that is used for co-locating and co-partitioning data from multiple datasets across nodes.

Example: trip_distance_mi

Attributes of Objects in the `partitionFields` Array

name String

Name of the field from the anchor dataset on which the rows in the Reflection are to be partitioned. If a column is listed as a partition column, it cannot also be listed as a sort column for the same Reflection. In aggregation Reflections, each column specified as a partition column or used in transform must also be listed as a dimension column. In raw Reflections, each column specified as a partition column or used in transform must also be listed as a display column.

Example: trip_distance_mi

transform Object

The type of partition transform that is applied. The value is an enum. The types are:

IDENTITY: Creates one partition per value. This is the default transform. If no transform is specified for a field named by the `name` property, an IDENTITY transform is

performed.

IDENTITY Example

```
{
  "name": "passenger_count",
  "transform": {
    "type": "IDENTITY"
  }
}
```

YEAR: Partitions by year. The field must use the TIMESTAMP or DATE data type.

YEAR Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "YEAR"
  }
}
```

MONTH: Partitions by month. The field must use the TIMESTAMP or DATE data type.

MONTH Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "MONTH"
  }
}
```

DAY: Partitions on the equivalent of dateint. The field must use the TIMESTAMP or DATE data type.

DAY Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "DAY"
  }
}
```

HOURL: Partitions on the equivalent of dateint and hour. The field must use the TIMESTAMP data type.

HOURL Example

```
{
  "name": "pickup_datetime",
```

```

"transform": {
  "type": "HOUR"
}
}

```

BUCKET: Partitions data into the number of partitions specified by an integer. For example, if the integer value N is specified, the data is partitioned into N, or (0 to (N-1)), partitions. The partition in which an individual row is stored is determined by hashing the field value and then calculating `<hash_value> mod N`. If the result is 0, the row is placed in partition 0; if the result is 1, the row is placed in partition 1; and so on.

This value must be followed by a ``bucketTransform`` object. This object takes one property: ``bucketCount``. This property takes an integer value.

BUCKET Example

```

{
  "name": "pickup_datetime",
  "transform": {
    "type": "BUCKET",
    "bucketTransform": {
      "bucketCount": 1000
    }
  }
}

```

TRUNCATE: If the specified field uses the string data type, truncates strings to a maximum of the number of characters specified by an integer. For example, suppose the specified transform is ``truncate(1, stateUS)``. A value of ``CA`` is truncated to ``C``, and the row is placed in partition C. A value of ``CO`` is also truncated to ``C``, and the row is also placed in partition C.

If the specified field uses the integer or long data type, truncates field values in the following way: For any ``truncate(L, col)``, truncates the field value to the biggest multiple of L that is smaller than the field value. For example, suppose the specified transform is ``truncate(10, intField)``. A value of 1 is truncated to 0 and the row is placed in the partition 0. A value of 247 is truncated to 240 and the row is placed in partition 240. If the transform is ``truncate(3, intField)``, a value of 13 is truncated to 12 and the row is placed in partition 12. A value of 255 is not truncated, because it is divisible by 3, and the row is placed in partition 255. This value must be followed by a ``truncateTransform`` object.

This object takes one property: ``truncateLength``. This property takes an integer value.

note

The truncate transform does not change field values. It uses field values to calculate the correct partitions in which to place rows.

TRUNCATE Example

```

{
  "name": "pickup_hour",

```

```
"transform": {
  "type": "TRUNCATE",
  "truncateTransform": {
    "truncateLength": 3
  }
}
```

Attributes of Objects in the `sortFields` Array

name String

Name of the field from the anchor dataset that is used for sorting in the Reflection.

Example: dropoff_date

Create a Reflection

Create a new Reflection.

Method and URL

POST /api/v3/reflection

Parameters

type Body String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name Body String

Name to use for the Reflection.

Example: New Aggregation Reflection

datasetId Body String (UUID)

Unique identifier of the anchor dataset to associate with the Reflection.

Example: df99ab32-c2d4-4d1c-9e91-2c8be861bb8a

enabled Body Boolean

If the Reflection should be available for accelerating queries, set to `true`. Otherwise, set to `false`.

Example: true

arrowCachingEnabled Body Boolean Optional

If Dremio should convert data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, set to `true`. Otherwise, set to `false` (default).

Example: false

displayFields Body Array of Object Optional

Information about the fields to display from the anchor dataset. The displayfields array must list every field in the anchor dataset or the Reflection fails. Each displayFields object contains one attribute: name. Valid only for raw Reflections.

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}, {"name": "trip_distance_mi"}, {"name": "fare_amount"}, {"name": "tip_amount"}, {"name": "total_amount"}]

dimensionFields Body Array of Object Optional

Information about the dimension fields from the anchor dataset to use in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each dimensionFields object contains two attributes: name and granularity. Valid only for aggregation Reflections.

Example: [{"name": "pickup_datetime", "granularity": "DATE"}, {"name": "passenger_count", "granularity": "DATE"}, {"name": "total_amount", "granularity": "DATE"}, {"name": "trip_distance_mi", "granularity": "DATE"}]

measureFields Body Array of Object Optional

Information about the measure fields from the anchor dataset to use in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each measureFields object contains two attributes: name and measureTypeList. Valid only for aggregation Reflections.

Example: [{"name": "passenger_count", "measureTypeList": ["SUM", "COUNT"]}, {"name": "trip_distance_mi", "measureTypeList": ["SUM", "COUNT"]}, {"name": "fare_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "tip_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "total_amount", "measureTypeList": ["SUM", "COUNT"]}]

distributionFields Body Array of Object Optional

Information about the distribution fields from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. Each distributionFields object contains one attribute: name.

Example: [{"name": "trip_distance_mi"}, {"name": "total_amount"}]

partitionFields Body Array of Object Optional

Information about the fields from the anchor dataset to use to partition data in the Reflection. Each field name is listed as an individual object. For more information, read [Horizontally Partition Reflections that Have Many Rows](#).

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}]

sortFields Body Array of Object Optional

Information about the fields from the anchor dataset to use for sorting in the Reflection. Each sortFields object contains one attribute: name. For more information, read [Sort Reflections on High-Cardinality Fields](#).

Example: [{"name": "trip_distance_mi"}]

partitionDistributionStrategy Body String Optional

Method to use to optimize data compression when executing Reflections. If set to `CONSOLIDATED` (default), Dremio minimizes the number of files produced. If set to `STRIPED`, Dremio minimizes the time required to refresh the Reflection.

Enum: CONSOLIDATED, STRIPED

Example: CONSOLIDATED

canView Body Boolean Optional

To view Reflections on all datasets of a source, system, space, or folder, set to `true` (default). Otherwise, set to `false`.

Example: true

canAlter Body Boolean Optional

To create, edit, and view Reflections on all datasets of a source, system, space, or folder, set to `true` (default). Otherwise, set to `false`.

Example: true

entityType Body String Optional

Type of the object. For Reflection objects, the entityType is `reflection`.

Example: reflection

Parameters of Objects in the `displayFields` Array

name Body String

Name of the field to display from the anchor dataset.

Example: "name": "pickup_datetime"

Parameters of Objects in the `dimensionFields` Array

name Body String

Name of the field from the anchor dataset to configure as a dimension for the Reflection.

Example: "name": "pickup_datetime"

granularity Body String

Grouping to use for the dimension field. If Dremio should automatically extract the day-level date value and use it as the grouping value in the Reflection, `DATE`. If Dremio should use the original value for grouping, `NORMAL`.

Enum: DATE, NORMAL

Example: "granularity": "DATE"

Parameters of Objects in the `measureFields` Array

name Body String

Name of the field from the anchor dataset that you expect to use in calculations. Fields of types `LIST`, `MAP`, and `UNION` are not valid measureFields.

Example: "name": "passenger_count"

measureTypeList Body Array of String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. In aggregation Reflections, every field listed as a distribution field must also be listed as a dimension field.

Enum: APPROX_COUNT_DISTINCT, MIN, MAX, UNKNOWN, SUM, COUNT

Example: ["SUM", "COUNT"]

Parameters of Objects in the `distributionFields` Array

name Body String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. In aggregation Reflections, every field listed as a distribution field must also be listed as a dimension field.

Example: "name": "trip_distance_mi"

Parameters of Objects in the `partitionFields` Array

name Body String

Name of the field from the anchor dataset on which you want to be able to partition rows. If you are creating an aggregation Reflection, every field listed as a partition field must also be listed as a dimension field. If you list a field as a partition field, you cannot list the same field as a sort field in the same Reflection.

Example: "name": "pickup_datetime"

transform Object

The type of partition transform that is applied. The value is an enum. The types are:

IDENTITY: Creates one partition per value. This is the default transform. If no transform is specified for a field named by the `name` property, an IDENTITY transform is performed.

IDENTITY Example

```
{
  "name": "passenger_count",
  "transform": {
    "type": "IDENTITY"
  }
}
```

YEAR: Partitions by year. The field must use the TIMESTAMP or DATE data type.

YEAR Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "YEAR"
  }
}
```

MONTH: Partitions by month. The field must use the TIMESTAMP or DATE data type.

MONTH Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "MONTH"
  }
}
```

DAY: Partitions on the equivalent of dateint. The field must use the TIMESTAMP or DATE data type.

DAY Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "DAY"
  }
}
```

HOURL: Partitions on the equivalent of dateint and hour. The field must use the TIMESTAMP data type.

HOURL Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "HOURL"
  }
}
```

```
}  
}
```

BUCKET: Partitions data into the number of partitions specified by an integer. For example, if the integer value N is specified, the data is partitioned into N, or (0 to (N-1)), partitions. The partition in which an individual row is stored is determined by hashing the field value and then calculating `<hash_value> mod N`. If the result is 0, the row is placed in partition 0; if the result is 1, the row is placed in partition 1; and so on.

This value must be followed by a ``bucketTransform`` object. This object takes one property: ``bucketCount``. This property takes an integer value.

BUCKET Example

```
{  
  "name": "pickup_datetime",  
  "transform": {  
    "type": "BUCKET",  
    "bucketTransform": {  
      "bucketCount": 1000  
    }  
  }  
}
```

TRUNCATE: If the specified field uses the string data type, truncates strings to a maximum of the number of characters specified by an integer. For example, suppose the specified transform is ``truncate(1, stateUS)``. A value of ``CA`` is truncated to ``C``, and the row is placed in partition C. A value of ``CO`` is also truncated to ``C``, and the row is also placed in partition C.

If the specified field uses the integer or long data type, truncates field values in the following way: For any ``truncate(L, col)``, truncates the field value to the biggest multiple of L that is smaller than the field value. For example, suppose the specified transform is ``truncate(10, intField)``. A value of 1 is truncated to 0 and the row is placed in the partition 0. A value of 247 is truncated to 240 and the row is placed in partition 240. If the transform is ``truncate(3, intField)``, a value of 13 is truncated to 12 and the row is placed in partition 12. A value of 255 is not truncated, because it is divisible by 3, and the row is placed in partition 255. This value must be followed by a ``truncateTransform`` object.

This object takes one property: ``truncateLength``. This property takes an integer value.

note

The truncate transform does not change field values. It uses field values to calculate the correct partitions in which to place rows.

TRUNCATE Example

```
{  
  "name": "pickup_hour",  
  "transform": {  
    "type": "TRUNCATE",
```

```

    "truncateTransform": {
      "truncateLength": 3
    }
  }
}

```

Parameters of Objects in the `sortFields` Array

name Body String

Name of the field from the anchor dataset to use for sorting in the Reflection. Every field listed as a sort field must also be listed as a dimension field. If you list a field as a sort field, you cannot list the same field as a partition field in the same Reflection.

Example: "name": "trip_distance_mi"

Example

Request

```

curl -X POST 'https://{hostname}/api/v3/reflection/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "type": "AGGREGATION",
  "name": "New Aggregation Reflection",
  "datasetId": "gc870df7-ddf7-4dle-bb9e-beef28ae773f",
  "enabled": true,
  "arrowCachingEnabled": false,
  "dimensionFields": [
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
}

```

```

    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    },
    {
      "name": "total_amount"
    }
  ],
  "partitionFields": [
    {
      "name": "pickup_datetime"
    },
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "entityType": "reflection"
}'

```

Response

```
{
  "id": "836eae91-306e-487b-a687-31c999653a86",
  "type": "AGGREGATION",
  "reflectionMode": "Manual",
  "name": "New Aggregation Reflection",
  "tag": "sEHieiuiinqE=",
  "createdAt": "2023-01-30T14:30:24.311Z",
  "updatedAt": "2023-01-30T14:30:24.311Z",
  "datasetId": "gc870df7-ddf7-4d1e-bb9e-beef28ae773f",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "NONE",
    "combinedStatus": "CANNOT_ACCELERATE_SCHEDULED",
    "failureCount": 0,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "dimensionFields": [
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  {
```

```

        "name": "fare_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "tip_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    }
],
"distributionFields": [
    {
        "name": "trip_distance_mi"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "pickup_datetime"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Response Status Codes

200 OK

- 401 Unauthorized
- 404 Not Found
- 405 Method Not Allowed
- 500 Internal Server Error

Retrieve All Reflections Enterprise

Retrieve a list of Reflection objects that includes all raw and aggregation Reflections in the Dremio instance.

Method and URL

```
GET /api/v3/reflection
```

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/reflection/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

In the response for a request to retrieve all raw and aggregation Reflections, the Reflection objects are wrapped with a data array. Each object in the data array represents one Reflection.

Response

```
{
  "data": [
    {
      "id": "95dda9dd-2371-467f-b68d-fc4c5ea57a8b",
      "type": "AGGREGATION",
      "name": "Aggregation Reflection",
      "tag": "ZpzGgxw2l04=",
      "createdAt": "2022-07-05T19:19:40.244Z",
      "updatedAt": "2023-01-10T17:12:40.244Z",
      "datasetId": "df99ab32-c2d4-4d1c-9e91-2c8be861bb8a",
      "currentSizeBytes": 18639885,
      "totalSizeBytes": 142639924,
      "enabled": true,
      "arrowCachingEnabled": false,
      "status": {
        "config": "OK",
        "refresh": "SCHEDULED",
        "availability": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "failureCount": 0,
        "lastDataFetch": "2023-01-10T17:12:40.244Z",
```

```

    "expiresAt": "3022-07-05T19:19:40.244Z"
  },
  "dimensionFields": [
    {
      "name": "pickup_date"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "dropoff_date"
    },
    {
      "name": "dropoff_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "surcharge",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ]
}

```

```

    ],
    {
      "name": "tip_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    },
    {
      "name": "total_amount"
    }
  ],
  "partitionFields": [
    {
      "name": "dropoff_date"
    },
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
},
{
  "id": "14f22052-cbb3-4d5d-8bbc-6154cca98e49",
  "type": "RAW",
  "name": "listings",
  "tag": "XAy4ccVFX04=",
  "createdAt": "2022-07-12T16:45:35.249Z",
  "updatedAt": "2022-07-12T16:45:35.249Z",
  "datasetId": "7707981c-cb33-42bc-a048-d27a8915f468",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,

```

```

"arrowCachingEnabled": true,
"status": {
  "config": "OK",
  "refresh": "MANUAL",
  "availability": "NONE",
  "combinedStatus": "CANNOT_ACCELERATE_MANUAL",
  "failureCount": 0,
  "lastDataFetch": "1969-12-31T23:59:59.999Z",
  "expiresAt": "1969-12-31T23:59:59.999Z"
},
"displayFields": [
  {
    "name": "id"
  }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
},
{
  "id": "6c209200-b522-4f81-bbe0-d10668c7752c",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "SQeEAG3d6DA=",
  "createdAt": "2021-09-29T15:47:44.806Z",
  "updatedAt": "2021-09-29T15:47:44.806Z",
  "datasetId": "746f867a-c27c-4711-bb8c-99546a4c25e0",
  "currentSizeBytes": 0,
  "totalSizeBytes": 1675978,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "GIVEN_UP",
    "availability": "NONE",
    "combinedStatus": "FAILED",
    "failureCount": 3,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "dimensionFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    }
  ],
  "measureFields": [
    {
      "name": "trip_distance_mi",
      "measureTypeList": [

```

```

        "COUNT",
        "SUM"
    ],
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    },
    {
        "name": "tip_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    },
    {
        "name": "fare_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
},
{
    "id": "c5c5b282-ffea-4a34-835f-cc591584412b",
    "type": "AGGREGATION",
    "name": "Test reflection",
    "tag": "lMxFcc2qjgE=",
    "createdAt": "2021-10-11T18:44:27.064Z",
    "updatedAt": "2021-10-11T18:44:27.064Z",
    "datasetId": "316531b8-3c56-42f2-b05f-81f228ef3162",
    "currentSizeBytes": 0,
    "totalSizeBytes": 0,
    "enabled": true,
    "arrowCachingEnabled": false,
    "status": {
        "config": "OK",
        "refresh": "MANUAL",
        "availability": "NONE",
        "combinedStatus": "CANNOT_ACCELERATE_MANUAL",
        "failureCount": 0,
        "lastDataFetch": "1969-12-31T23:59:59.999Z",
        "expiresAt": "1969-12-31T23:59:59.999Z"
    },
    "dimensionFields": [
        {

```

```

        "name": "passenger_count"
    }
],
"measureFields": [
    {
        "name": "trip_distance_mi",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    },
    {
        "name": "tip_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    },
    {
        "name": "fare_amount",
        "measureTypeList": [
            "COUNT",
            "SUM"
        ]
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}
],
"canAlterReflections": true
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieve a Reflection

Retrieve the specified Reflection.

Method and URL

```
GET /api/v3/reflection/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the Reflection that you want to retrieve.

Example: 95dda9dd-2371-467f-b68d-fc4c5ea57a8b

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/reflection/95dda9dd-2371-467f-b68d-fc4c5ea57a8b'

--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "95dda9dd-2371-467f-b68d-fc4c5ea57a8b",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "ZpzGgxw2l04=",
  "createdAt": "2022-07-05T19:19:40.244Z",
  "updatedAt": "2023-01-10T17:12:40.244Z",
  "datasetId": "df99ab32-c2d4-4d1c-9e91-2c8be861bb8a",
  "currentSizeBytes": 18639885,
  "totalSizeBytes": 142639924,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "failureCount": 0,
    "lastDataFetch": "2023-01-10T17:12:40.244Z",
    "expiresAt": "3022-07-05T19:19:40.244Z"
  },
  "dimensionFields": [
    {
      "name": "pickup_date"
```

```

    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "dropoff_date"
    },
    {
      "name": "dropoff_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "surcharge",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [

```



```

        "SUM",
        "COUNT"
    ],
    },
    {
        "name": "total_amount",
        "measureTypeList": [
            "SUM",
            "COUNT"
        ]
    }
],
"distributionFields": [
    {
        "name": "trip_distance_mi"
    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "dropoff_date"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieve All Reflections for a Dataset

Retrieve all raw and aggregation Reflections for the specified dataset.

Method and URL

```
GET /api/v3/dataset/{datasetId}/reflection
```

Parameters

datasetId Path String (UUID)

Unique identifier of the dataset whose Reflections you want to retrieve.

Example: 3cbab7b3-ee82-44c1-abcc-e86d56078d4d

Example

Request

```
curl -X GET \
'https://{hostname}/api/v3/dataset/3cbab7b3-ee82-44c1-abcc-e86d56078d4d/reflection' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

In the response for a request to retrieve all Reflections for a dataset, the Reflection objects are wrapped with a data array. Each object in the data array represents one Reflection.

Response

```
{
  "data": [
    {
      "id": "23f75eb1-045f-447f-b3fa-374377877569",
      "type": "RAW",
      "name": "Raw Reflection",
      "tag": "K9J2SHE0c+Q=",
      "createdAt": "2023-02-03T16:38:27.770Z",
      "updatedAt": "2023-02-03T16:38:27.770Z",
      "datasetId": "3cbab7b3-ee82-44c1-abcc-e86d56078d4d",
      "currentSizeBytes": 0,
      "totalSizeBytes": 0,
      "enabled": true,
      "arrowCachingEnabled": false,
      "status": {
        "config": "OK",
        "refresh": "MANUAL",
        "availability": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "failureCount": 0,
        "lastDataFetch": "2023-02-03T16:38:27.780Z",
        "expiresAt": "3022-06-06T16:38:27.780Z"
      }
    },
  ],
}
```

```

"displayFields": [
  {
    "name": "pickup_datetime"
  },
  {
    "name": "passenger_count"
  },
  {
    "name": "trip_distance_mi"
  },
  {
    "name": "fare_amount"
  },
  {
    "name": "tip_amount"
  },
  {
    "name": "total_amount"
  }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
},
{
  "id": "3cbab7b3-ee82-44c1-abcc-e86d56078d4d",
  "type": "AGGREGATION",
  "name": "Aggregation Reflection",
  "tag": "Mc4hDFk5JR8=",
  "createdAt": "2023-02-03T16:39:40.556Z",
  "updatedAt": "2023-02-03T16:39:40.556Z",
  "datasetId": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "MANUAL",
    "availability": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "failureCount": 0,
    "lastDataFetch": "2023-02-03T16:39:40.568Z",
    "expiresAt": "3022-06-06T16:39:40.568Z"
  },
  "dimensionFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    }
  ]
}

```

```

],
  "measureFields": [
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "COUNT",
        "SUM"
      ]
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
},
"canAlterReflections": true
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Refresh a Reflection

For information about the refresh action performed, see [Triggering Refreshes by Using the Reflection API, the Catalog API, or an SQL Command](#).

Method and URL

```
POST /api/v3/reflection/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier for the Reflection that you want to base the refresh action on.

Example: 836eae91-306e-487b-a687-31c999653a86

Response Status Codes

200 OK

400 Not supported

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Update a Reflection

Update the specified Reflection.

Method and URL

```
PUT /api/v3/reflection/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the Reflection that you want to update.

Example: 836eae91-306e-487b-a687-31c999653a86

type Body String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name Body String

Name to use for the Reflection.

Example: New Aggregation Reflection

tag Body String

Unique identifier of the most recent version of the Reflection. Dremio uses the tag to ensure that you are updating the most recent version of the Reflection.

Example: ZpzGgxw2l04=

datasetId Body String (UUID)

Unique identifier of the anchor dataset associated with the Reflection.

Example: gc870df7-ddf7-4d1e-bb9e-beef28ae773f

enabled Body Boolean

If the Reflection should be available for accelerating queries, set to `true`. Otherwise, set to `false`.

Example: false

arrowCachingEnabled Body Boolean Optional

If Dremio should convert data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, set to `true`. Otherwise, set to `false` (default).

Example: true

displayFields Body Array of Object

Information about the fields to display from the anchor dataset. The displayfields array must list every field in the anchor dataset or the Reflection fails. Each displayFields object contains one attribute: name. Valid only for raw Reflections.

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}, {"name": "trip_distance_mi"}, {"name": "fare_amount"}, {"name": "tip_amount"}, {"name": "total_amount"}]

dimensionFields Body Array of Object

Information about the dimension fields from the anchor dataset to use in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Each dimensionFields object contains two attributes: name and granularity. Valid only for aggregation Reflections. If you omit the dimensionFields object in a PUT request, Dremio removes all existing dimension fields from the Reflection. To keep existing dimension fields while making other updates, duplicate the existing dimensionFields array in the PUT request.

Example: [{"name": "pickup_datetime","granularity": "DATE"}, {"name": "passenger_count","granularity": "DATE"}, {"name": "total_amount","granularity": "DATE"}, {"name": "trip_distance_mi","granularity": "DATE"}]

measureFields Body Array of Object

Information about the measure fields from the anchor dataset to use in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Each measureFields object contains two attributes: name and measureTypeList. Valid only for aggregation Reflections. If you omit the measureFields object in a PUT request, Dremio removes all existing measure fields from the Reflection. To keep existing measure fields while making other updates, duplicate the existing measureFields array in the PUT request.

Example: [{"name": "passenger_count","measureTypeList": ["SUM", "COUNT"]}, {"name": "trip_distance_mi","measureTypeList": ["SUM", "COUNT"]}, {"name": "fare_amount","measureTypeList": ["SUM", "COUNT"]}, {"name": "tip_amount","measureTypeList": ["SUM", "COUNT"]}, {"name": "total_amount","measureTypeList": ["SUM", "COUNT"]}]

distributionFields Body Array of Object Optional

Information about the distribution fields from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. Each distributionFields object contains one attribute: name.

If you omit the distributionFields object in a PUT request, Dremio removes all existing distribution fields from the Reflection. To keep existing distribution fields while making other updates, duplicate the existing distributionFields array in the PUT request.

Example: [{"name": "trip_distance_mi"}, {"name": "total_amount"}]

partitionFields Body Array of Object Optional

Information about the fields from the anchor dataset to use to partition data in the Reflection. Each field name is listed as an individual object. If you omit the partitionFields object in a PUT request, Dremio removes all existing partition fields from the Reflection. To keep existing partition fields while making other updates, duplicate the existing partitionFields array in the PUT request. For more information, read [Horizontally Partition Reflections that Have Many Rows](#).

Example: [{"name": "pickup_datetime"}, {"name": "passenger_count"}]

sortFields Body Array of Object

Information about the fields from the anchor dataset to use for sorting in the Reflection. Each sortFields object contains one attribute: name. If you omit the sortFields object in a PUT request, Dremio removes all existing sort fields from the Reflection. To keep existing sort fields while making other updates, duplicate the existing sortFields array in the PUT request. For more information, read [Sort Reflections on High-Cardinality Fields](#).

Example: "name": "trip_distance_mi"

partitionDistributionStrategy Body String Optional

Method to use to optimize data compression when executing Reflections. If set to `CONSOLIDATED` (default), Dremio minimizes the number of files produced. If set to `STRIPED`, Dremio minimizes the time required to refresh the Reflection.

Enum: CONSOLIDATED, STRIPED

Example: CONSOLIDATED

Parameters of Objects in the `displayFields` Array

name Body String

Name of the field to display from the anchor dataset.

Example: "name": "pickup_datetime"

Parameters of Objects in the `dimensionFields` Array

name Body String

Name of the field from the anchor dataset to configure as a dimension for the Reflection.

Example: "name": "pickup_datetime"

granularity Body String

Grouping to use for the dimension field. If Dremio should automatically extract the day-level date value and use it as the grouping value in the Reflection, `DATE`. If Dremio should use the original value for grouping, `NORMAL`.

Enum: DATE, NORMAL

Example: "granularity": "DATE"

Parameters of Objects in the `measureFields` Array

name Body String

Name of the field from the anchor dataset that you expect to use in calculations. Fields of types `LIST`, `MAP`, and `UNION` are not valid measureFields.

Example: "name": "passenger_count"

measureTypeList Body Array of String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. Every field listed as a distribution field must also be listed as a dimension field.

Enum: APPROX_COUNT_DISTINCT, MIN, MAX, UNKNOWN, SUM, COUNT

Example: ["SUM", "COUNT"]

Parameters of Objects in the `distributionFields` Array

name Body String

Name of the field from the anchor dataset to use for co-locating and co-partitioning data from multiple datasets across nodes. In aggregation Reflections, every field listed as a distribution field must also be listed as a dimension field.

Example: "name": "pickup_datetime"

Parameters of Objects in the `partitionFields` Array

name Body String

Name of the field from the anchor dataset on which the rows in the Reflection are to be partitioned. If a column is listed as a partition column, it cannot also be listed as a sort column for the same Reflection. In aggregation Reflections, each column specified as a partition column or used in transform must also be listed as a dimension column. In raw Reflections, each column specified as a partition column or used in transform must also be listed as a display column.

Example: "name": "dropoff_date"

transform Object

The type of partition transform that is applied. The value is an enum. The types are:

IDENTITY: Creates one partition per value. This is the default transform. If no transform is specified for a field named by the `name` property, an IDENTITY transform is performed.

IDENTITY Example

```
{
  "name": "passenger_count",
  "transform": {
    "type": "IDENTITY"
  }
}
```

YEAR: Partitions by year. The field must use the TIMESTAMP or DATE data type.

YEAR Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "YEAR"
  }
}
```

MONTH: Partitions by month. The field must use the TIMESTAMP or DATE data type.

MONTH Example

```
{
  "name": "pickup_datetime",
  "transform": {
```

```
    "type": "MONTH"
  }
}
```

DAY: Partitions on the equivalent of dateint. The field must use the TIMESTAMP or DATE data type.

DAY Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "DAY"
  }
}
```

HOURL: Partitions on the equivalent of dateint and hour. The field must use the TIMESTAMP data type.

HOURL Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "HOURL"
  }
}
```

BUCKET: Partitions data into the number of partitions specified by an integer. For example, if the integer value N is specified, the data is partitioned into N, or (0 to (N-1)), partitions. The partition in which an individual row is stored is determined by hashing the field value and then calculating `<hash_value> mod N`. If the result is 0, the row is placed in partition 0; if the result is 1, the row is placed in partition 1; and so on.

This value must be followed by a `bucketTransform` object. This object takes one property: `bucketCount`. This property takes an integer value.

BUCKET Example

```
{
  "name": "pickup_datetime",
  "transform": {
    "type": "BUCKET",
    "bucketTransform": {
      "bucketCount": 1000
    }
  }
}
```

TRUNCATE: If the specified field uses the string data type, truncates strings to a

maximum of the number of characters specified by an integer. For example, suppose the specified transform is ``truncate(1, stateUS)``. A value of ``CA`` is truncated to ``C``, and the row is placed in partition C. A value of ``CO`` is also truncated to ``C``, and the row is also placed in partition C.

If the specified field uses the integer or long data type, truncates field values in the following way: For any ``truncate(L, col)``, truncates the field value to the biggest multiple of L that is smaller than the field value. For example, suppose the specified transform is ``truncate(10, intField)``. A value of 1 is truncated to 0 and the row is placed in the partition 0. A value of 247 is truncated to 240 and the row is placed in partition 240. If the transform is ``truncate(3, intField)``, a value of 13 is truncated to 12 and the row is placed in partition 12. A value of 255 is not truncated, because it is divisible by 3, and the row is placed in partition 255. This value must be followed by a ``truncateTransform`` object.

This object takes one property: ``truncateLength``. This property takes an integer value.

note

The truncate transform does not change field values. It uses field values to calculate the correct partitions in which to place rows.

TRUNCATE Example

```
{
  "name": "pickup_hour",
  "transform": {
    "type": "TRUNCATE",
    "truncateTransform": {
      "truncateLength": 3
    }
  }
}
```

Parameters of Objects in the ``sortFields`` Array

name Body String

Name of the field from the anchor dataset to use for sorting in the Reflection. Every field listed as a sort field must also be listed as a dimension field. If you list a field as a sort field, you cannot list the same field as a partition field in the same Reflection.

Example: "name": "pickup_datetime"

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/reflection/836eae91-306e-487b-a687-31c999653a86' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
```

```

"id": "836eae91-306e-487b-a687-31c999653a86",
"type": "AGGREGATION",
"name": "New Aggregation Reflection",
>tag": "sEHieiuiinqE=",
"datasetId": "gc870df7-ddf7-4dle-bb9e-beef28ae773f",
"enabled": false,
"arrowCachingEnabled": true,
"dimensionFields": [
  {
    "name": "pickup_datetime",
    "granularity": "DATE"
  },
  {
    "name": "passenger_count"
  },
  {
    "name": "total_amount"
  },
  {
    "name": "trip_distance_mi"
  }
],
"measureFields": [
  {
    "name": "passenger_count",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "trip_distance_mi",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "fare_amount",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "tip_amount",
    "measureTypeList": [
      "SUM",
      "COUNT"
    ]
  },
  {
    "name": "total_amount",
    "measureTypeList": [

```

```

        "SUM",
        "COUNT"
    ]
  },
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    },
    {
      "name": "total_amount"
    }
  ],
  "partitionFields": [
    {
      "name": "pickup_datetime"
    },
    {
      "name": "passenger_count"
    }
  ],
  "sortFields": [
    {
      "name": "trip_distance_mi"
    }
  ],
  "entityType": "reflection"
}'

```

Response

```

{
  "id": "836eae91-306e-487b-a687-31c999653a86",
  "type": "AGGREGATION",
  "name": "New Aggregation Reflection",
  "tag": "nRPbilwodqC=",
  "createdAt": "2023-01-30T14:35:19.192Z",
  "updatedAt": "2023-01-30T14:35:19.192Z",
  "datasetId": "gc870df7-ddf7-4dle-bb9e-beef28ae773f",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": false,
  "arrowCachingEnabled": true,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "NONE",
    "combinedStatus": "DISABLED",
    "failureCount": 0,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "dimensionFields": [

```

```

    {
      "name": "pickup_datetime",
      "granularity": "DATE"
    },
    {
      "name": "passenger_count"
    },
    {
      "name": "total_amount"
    },
    {
      "name": "trip_distance_mi"
    }
  ],
  "measureFields": [
    {
      "name": "passenger_count",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "trip_distance_mi",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "fare_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "tip_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    },
    {
      "name": "total_amount",
      "measureTypeList": [
        "SUM",
        "COUNT"
      ]
    }
  ],
  "distributionFields": [
    {
      "name": "trip_distance_mi"
    }
  ]
}

```

```

    },
    {
        "name": "total_amount"
    }
],
"partitionFields": [
    {
        "name": "pickup_datetime"
    },
    {
        "name": "passenger_count"
    }
],
"sortFields": [
    {
        "name": "trip_distance_mi"
    }
],
"partitionDistributionStrategy": "CONSOLIDATED",
"canView": true,
"canAlter": true,
"entityType": "reflection"
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Delete a Reflection

Delete the specified Reflection.

Method and URL

```
DELETE /api/v3/reflection/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the Reflection that you want to delete.

Example: 95dda9dd-2371-467f-b68d-fc4c5ea57a8b

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/reflection/95dda9dd-2371-467f-b68d-fc4c5ea57a8b' --header 'Authorization: Bearer <dremioAccessToken>' \ --header 'Content-Type:application/json'
```

Response

No response

Response Status Codes

- 200 OK
- 401 Unauthorized
- 404 Not Found
- 405 Method Not Allowed

Was this page helpful?

[Previous

Personal Access Token](/current/reference/api/personal-access-token)[Next Recommendations](/current/reference/api/reflections/reflection-recommendations)

Reflection Attributes

Create a Reflection

Parameters

Example

Response Status Codes

Retrieve All Reflections Enterprise

Example

Response Status Codes

Retrieve a Reflection

Parameters

Example

Response Status Codes

Retrieve All Reflections for a Dataset

Parameters

Example

Response Status Codes

Refresh a Reflection

Parameters

Response Status Codes

Update a Reflection

Parameters

Example

Response Status Codes

Delete a Reflection

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/roles/>

Version: current [26.x]

On this page

Role Enterprise

Use the Role API to manage roles.

Role Object

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
      "name": "qa_team1",
      "type": "INTERNAL"
    }
  ],
}
```

```

    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 3,
  "description": "Role for testing the new feature"
}

```

Role Attributes

id String (UUID)

Unique identifier of the role.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

name String

User-provided name of the role.

Example: Temporary Testing

type String

Origin of the role.

~INTERNAL~: Role was created in the Dremio user interface (UI) or with the Role API.

~EXTERNAL~: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

~SYSTEM~: Role was predefined in Dremio.

Example: INTERNAL

roles Array of Object

Information about the roles to which the role belongs.

Example: [{"id": "6f87a9c5-d733-4935-8331-875a4a8e09d7", "name": "SELECT and CREATE", "type": "INTERNAL"}, {"id": "f8426061-8413-46ec-a84d-1b481a97b248", "name": "VIEW", "type": "INTERNAL"}]

memberCount Integer

Number of users and roles that are members of the role.

Example: 3

description String

User-provided description of the role.

Example: Role for testing the new feature

Attributes of Objects in the `roles` Array

id String (UUID)

Unique identifier of the role.

Example: 6f87a9c5-d733-4935-8331-875a4a8e09d7

name String

Name of the role.

Example: SELECT and CREATE

type String

Origin of the role.

`INTERNAL`: Role was created in the Dremio user interface (UI) or with the Role API.

`EXTERNAL`: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

`SYSTEM`: Role was predefined in Dremio.

Example: INTERNAL

List Roles

Method and URL

```
GET /api/v3/role
```

Parameters

filter Query Object Optional

Filter the role list:

Role Fields	Functions	Operators
* `id` * `name` * `type`	* `startsWith()` * `contains()`	* `==` encoded as
`%3D%3D` (equals) * `&&` encoded as `%26%26` (logical and) * `||` encoded as
`%7C%7C` (logical or) |

The filter parameter value must be URL-encoded. Examples:

`?filter=name%3D%3D%27admin%27` evaluates to "name=='admin'"

`?filter=name%3D%3D%27Analyst%27%26%26type%3D%3D%27INTERNAL%27`
evaluates to "name=='Analyst'&&type=='INTERNAL'"

`?filter=name%2Econtains%28%27data%27%29%26%26type%3D%3D%27INTERNAL%27` evaluates to "name.contains('data')&&type=='INTERNAL'

For more information, see [filter Query Parameter](#).

Example

Request

```
curl -X GET "https://{hostname}/api/v3/role?filter=name%2Econtains%28%27data%27%29%26%26type%3D%3D%27INTERNAL%27" \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "id": "61fe3576-a8c2-4006-ae16-a414eb3cd29a",
      "name": "DataAnalyst",
      "type": "INTERNAL",
      "roles": [],
      "memberCount": 0,
      "description": ""
    }
  ],
  "totalResults": 1
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Create a Role

Method and URL

POST /api/v3/role

Parameters

name Body String

Name for the role. The role name must be unique and cannot be updated after the role is created.

Example: Temporary Testing

roles Body Array of Object Optional

Information about the roles to which the role should be assigned.

Example: [{"id": "6f87a9c5-d733-4935-8331-875a4a8e09d7"}, {"id": "f8426061-8413-46ec-a84d-1b481a97b248"}]

description Body String Optional

Description for the role.

Example: Role for testing the new feature

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role to which the role you create should be assigned.

Example: 6f87a9c5-d733-4935-8331-875a4a8e09d7

name Body String Optional

Name of the role to which the role you create should be assigned.

Example: qa_team1

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/role' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "Temporary Testing",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248"
    }
  ],
  "description": "Role for testing the new feature"
}'
```

Response

```
{
```

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
      "name": "qa_team1",
      "type": "INTERNAL"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 0,
  "description": "Role for testing the new feature"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

405 Method Not Allowed

Retrieve a Role by ID

Retrieve a specific role by the role's ID.

Method and URL

```
GET /api/v3/role/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the role you want to retrieve.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
      "name": "qa_team1",
      "type": "INTERNAL"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 3,
  "description": "Role for testing the new feature"
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieve a Role by Name

Use this command to import a role from your integrated identity provider (IdP) by specifying its name.

Method and URL

```
GET /api/v3/role/by-name/{name}
```

Parameters

name Path String

Name of the role you want to retrieve. The role name is case-insensitive. If the role name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Temporary%20Testing

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/role/by-name/Temporary%20Testing' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
      "name": "qa_team1",
      "type": "INTERNAL"
    },
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 3,
  "description": "Role for testing the new feature"
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Update a Role

Update the specified role.

Method and URL

PUT /api/v3/role/{id}

Parameters

id Path String (UUID)

Unique identifier of the role you want to update.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

id Body String (UUID)

Unique identifier of the role you want to update.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

name Body String

Name of the role.

Example: Temporary Testing

roles Body Array of Object Optional

Information about the roles to which the role should be assigned. If you omit an existing role in a PUT request, Dremio removes the role. To keep all existing roles while making other updates, include all existing roles in the PUT request.

Example: [{"id": "f8426061-8413-46ec-a84d-1b481a97b248"}]

description Body String Optional

Description to use for the role. If you omit the description in a PUT request, Dremio removes the existing description. To keep the existing description while making other updates, include the description in the PUT request.

Example: Role for viewing the new feature

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role to which the role you update should be assigned.

Example: f8426061-8413-46ec-a84d-1b481a97b248

name Body String Optional

Name of the role to which the role you update should be assigned.

Example: prod_testing

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "roles": [
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248"
    }
  ],
  "description": "Role for viewing the new feature"
}'
```

Response

```
{
  "id": "957a8af5-9211-4bc5-9fe5-1a44ff30304d",
  "name": "Temporary Testing",
  "type": "INTERNAL",
  "roles": [
    {
      "id": "f8426061-8413-46ec-a84d-1b481a97b248",
      "name": "prod_testing",
      "type": "INTERNAL"
    }
  ],
  "memberCount": 3,
  "description": "Role for viewing the new feature"
}
```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 404 Not Found
- 405 Method Not Allowed

Add and Remove Role Members

Add and remove members (roles and users) of the specified role.

Method and URL

```
PATCH /api/v3/role/{id}/member
```

Parameters

id Path String (UUID)

Unique identifier of the role for which you want to add or remove members.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

op Body String

Action to take for the user or role.

Enum: add, remove

Example: add

id Body String (UUID)

Unique identifier of the user or role to add or remove.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

type Body String

Type of member you want to add or remove.

Enum: role, user

Example: role

The request body is an array of objects. Each object includes the three parameters for a single user or role that you want to add or remove:

Example

Request

```
curl -X PATCH 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d/member' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '[
{
  "op": "add",
  "id": "f8426061-8413-46ec-a84d-1b481a97b248",
  "type": "role"
},
{
  "op": "add",
  "id": "671cdeb8-1af9-45b6-98ee-8ca1e0543a38",
  "type": "user"
},
{
  "op": "remove",
  "id": "6f87a9c5-d733-4935-8331-875a4a8e09d7",
  "type": "role"
}]'
```

```
},
{
  "op": "remove",
  "id": "614a6938-7a69-4f7c-ab96-00b50addb1f9",
  "type": "user"
}
]
```

Response

No response

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

405 Method Not Allowed

Delete a Role

Delete the specified role.

Method and URL

```
DELETE /api/v3/role/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the role that you want to delete.

note

It is not possible to delete a system role, like ADMIN or PUBLIC. Requests to delete a system role result in a `404 Not Found` response.

Example: 957a8af5-9211-4bc5-9fe5-1a44ff30304d

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/role/957a8af5-9211-4bc5-9fe5-1a44ff30304d' \
```

```
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Reflection](/current/reference/api/reflections/)[Next

Role Privileges](/current/reference/api/roles/privilege)

Role Attributes

List Roles

Parameters

Example

Response Status Codes

Create a Role

Parameters

Example

Response Status Codes

Retrieve a Role by ID

Parameters

Example

Response Status Codes

Retrieve a Role by Name

Parameters

-

Example

Response Status Codes

Update a Role

Parameters

Example

Response Status Codes

Add and Remove Role Members

Parameters

Example

Response Status Codes

Delete a Role

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/scripts/>

Version: current [26.x]

On this page

Scripts

Use the Scripts API to manage [scripts](#), retrieve the contents of scripts for use in a scheduler, retrieve and update privileges on scripts, and migrate scripts between different environments.

Scripts Object

```
{
  "total": 3,
  "data": [
    {
      "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
      "name": "newScript",
      "content": "SELECT * FROM sampled",
      "context": [
        "@dremio",
        "scriptsFolder"
      ],
      "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
    }
  ]
}
```

```

    "createdAt": "2024-05-24T17:42:00.304Z",
    "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
    "modifiedAt": "2024-05-24T17:42:00.304Z",
    "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
  },
  {
    "id": "37dab994-3f1f-4de7-b2e7-49cb5ff0b395",
    "name": "tmp_testing_04-15-24",
    "content": "-- create table accounting_storage.\"tmp.dremio.com\".test1(id
INT);\n-- refresh dataset test1\nalter table test1 REFRESH METADATA;",
    "context": [
      "accounting_storage",
      "tmp.dremio.com"
    ],
    "owner": "ba92bf87-174d-422e-becb-d526757c8099",
    "createdAt": "2024-04-15T10:38:31.433Z",
    "createdBy": "ba92bf87-174d-422e-becb-d526757c8099",
    "modifiedAt": "2024-04-15T11:03:27.542Z",
    "modifiedBy": "ba92bf87-174d-422e-becb-d526757c8099"
  },
  {
    "id": "02fef13e-cedd-46ac-b5bf-abcdcd092146",
    "name": "pop10000_A",
    "content": "SELECT * FROM Samples.\"samples.dremio.com\".\"\"zips.json\" WHERE pop >
10000 AND STARTS_WITH(city, 'A');\n\nCREATE OR REPLACE VIEW myView AS\nSELECT city,
state, pop FROM Samples.\"samples.dremio.com\".\"\"zips.json\"\"\nWHERE pop >
10000;\n\nCREATE OR REPLACE VIEW myView2 AS\nSELECT * FROM myView\nWHERE
STARTS_WITH(city, 'A');\n\nALTER TABLE myView2 \nCREATE RAW Reflection myReflection
\nUSING DISPLAY(city, state, pop);\n\nSELECT * FROM myView2;\n\nCREATE OR REPLACE
FUNCTION isMA(state VARCHAR)\n      RETURNS BOOLEAN\n      RETURN SELECT state =
'MA';\n\nALTER TABLE myView ADD ROW ACCESS POLICY isMA(\"state\");\n\nSELECT * FROM
myView2;\n\n// Refresh the reflection\n\nSELECT * FROM myView2;",
    "context": [
      "@dev"
    ],
    "owner": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
    "createdAt": "2024-05-16T18:08:06.363Z",
    "createdBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
    "modifiedAt": "2024-05-16T18:31:22.593Z",
    "modifiedBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921"
  }
]
}

```

Scripts Attributes

total Integer

Total number of scripts in the organization.

Example: 3

data Array of Object

List of the scripts in the organization, with an individual object representing each script.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the script. Generated by Dremio and immutable.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

name String

User-provided name of the script.

Example: newScript

content String

The script's SQL.

Example: SELECT * FROM sampled

context Array of String

Path where the SQL query specified in the content attribute runs. If no context is specified for the script, the value is an empty array.

Example: ["@dremio","scriptsFolder"]

owner String

Unique identifier for the user who owns the script.

Example: 8be516f3-04c4-4d19-824d-5a70b3c4442e

createdAt String

Date and time that the script was created. In UTC format.

Example: 2024-05-24T17:42:00.304Z

createdBy String

The unique identifier for the user who created the script.

Example: a97c694f-1e55-4f34-91aa-97f99fee802e

modifiedAt String

Date and time that the script was last modified. In UTC format.

Example: 2024-05-24T17:42:00.304Z

modifiedBy String

The unique identifier for the user who last modified the script.

Example: a97c694f-1e55-4f34-91aa-97f99fee802e

Create a Script

Create a script.

Method and URL

```
POST /api/v3/scripts
```

Parameters

name Body String

Name to use for the script.

Example: newScript

content Body String

The SQL for the script.

Example: SELECT * FROM sampledb

context Body Array of String Optional

Path where the SQL query specified in the content attribute should run.

Example: ["@dremio","scriptsFolder"]

owner Body String Optional

Unique identifier for the user who should own the script. Default is the ID for the user who creates the script.

Example: 8be516f3-04c4-4d19-824d-5a70b3c4442e

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/scripts' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "newScript",
  "content": "SELECT * FROM sampledb",
  "context": [
    "@dremio",
    "scriptsFolder"
  ],
  "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e"
}'
```

Response

```
{
  "id": "f873a72e-12a5-4537-a393-f9675da7c5f8",
  "name": "newScript",
  "content": "SELECT * FROM sampled",
  "context": [
    "@dremio",
    "scriptsFolder"
  ],
  "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
  "createdAt": "2024-05-24T17:42:00.304Z",
  "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "modifiedAt": "2024-05-24T17:42:00.304Z",
  "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

Retrieve All Scripts

Retrieve a list of all scripts in the organization.

Method and URL

```
GET /api/v3/scripts/
```

Parameters

createdBy Query String Optional

Unique identifier for the user who created the scripts you want to retrieve. For more information, read [createdBy Query Parameter](#).

maxResults Query Integer Optional

Maximum number of scripts to return in the response. Maximum valid value is `100`. Default is `25`. For more information, read [maxResults Query Parameter](#).

offset Query Integer Optional

Number of rows to skip for pagination. Default is `0`. Read [limit and offset Query Parameters](#) for usage examples.

orderBy Query String Optional

Organize the response in ascending (default) or descending order by name, createdAt, or modifiedAt. To specify descending order, precede the orderBy value with a '-' character. To organize the response by more than one attribute, use a comma-separated list: `?orderBy=createdAt,name`. For more information, read [orderBy Query Parameter](#).

ownedBy Query String Optional

Unique identifier for the user who owns the scripts you want to retrieve. For more information, read [ownedBy Query Parameter](#).

search Query String Optional

The string for which to search the values of the name attributes in scripts. Read [search Query Parameter](#) for usage examples.

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/scripts' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "total": 3,
  "data": [
    {
      "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
      "name": "newScript",
      "content": "SELECT * FROM sampled",
      "context": [
        "@dremio",
        "scriptsFolder"
      ],
      "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "createdAt": "2024-05-24T17:42:00.304Z",
      "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
      "modifiedAt": "2024-05-24T17:42:00.304Z",
      "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
    },
    {
      "id": "37dab994-3f1f-4de7-b2e7-49cb5ff0b395",
      "name": "tmp_testing_04-15-24",
      "content": "-- create table accounting_storage.\"tmp.dremio.com\".test1(id INT);\n-- refresh dataset test1\nalter table test1 REFRESH METADATA;",
      "context": [
        "accounting_storage",
        "tmp.dremio.com"
      ],
      "owner": "ba92bf87-174d-422e-becb-d526757c8099",
      "createdAt": "2024-04-15T10:38:31.433Z",
```

```

    "createdBy": "ba92bf87-174d-422e-becb-d526757c8099",
    "modifiedAt": "2024-04-15T11:03:27.542Z",
    "modifiedBy": "ba92bf87-174d-422e-becb-d526757c8099"
  },
  {
    "id": "02fef13e-cedd-46ac-b5bf-abcdcd092146",
    "name": "pop10000_A",
    "content": "SELECT * FROM Samples.\"samples.dremio.com\".\"\"zips.json\" WHERE pop > 10000 AND STARTS_WITH(city, 'A');\n\nCREATE OR REPLACE VIEW myView AS\nSELECT city, state, pop FROM Samples.\"samples.dremio.com\".\"\"zips.json\" WHERE pop > 10000;\n\nCREATE OR REPLACE VIEW myView2 AS\nSELECT * FROM myView\nWHERE STARTS_WITH(city, 'A');\n\nALTER TABLE myView2 \nCREATE RAW Reflection myReflection\nUSING DISPLAY(city, state, pop);\n\nSELECT * FROM myView2;\n\nCREATE OR REPLACE FUNCTION isMA(state VARCHAR)\n      RETURNS BOOLEAN\n      RETURN SELECT state = 'MA';\n\nALTER TABLE myView ADD ROW ACCESS POLICY isMA(\"state\");\n\nSELECT * FROM myView2;\n\n// Refresh the reflection\n\nSELECT * FROM myView2;",
    "context": [
      "@dev"
    ],
    "owner": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
    "createdAt": "2024-05-16T18:08:06.363Z",
    "createdBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921",
    "modifiedAt": "2024-05-16T18:31:22.593Z",
    "modifiedBy": "7a92baf7-646a-4bc5-b0f4-eaf18d0a9921"
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

Retrieve a Script by ID

Retrieve the specified script.

Method and URL

```
GET /api/v3/scripts/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the script you want to retrieve.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
  "name": "newScript",
  "content": "SELECT * FROM sampled",
  "context": [
    "@dremio",
    "scriptsFolder"
  ],
  "owner": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
  "createdAt": "2024-05-24T17:42:00.304Z",
  "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "modifiedAt": "2024-05-24T17:42:00.304Z",
  "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Update a Script

Update the specified script.

Method and URL

```
PATCH /api/v3/scripts/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the script.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

name Body String Optional

User-provided name to use for the script.

Example: updatedScript

content Body Array of String Optional

The updated SQL for the script.

Example: SELECT * FROM Samples."samples.dremio.com"."zips.json"

context Body String Optional

Path where the SQL query specified in the content attribute should run.

Example: ["@dremio","secondScriptsFolder"]

owner Body String Optional

Unique identifier for the user who should own the script.

Example: a97c694f-1e55-4f34-91aa-97f99fee802e

Example

Request

```
curl -X PATCH 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "updatedScript",
  "content": "SELECT * FROM Samples."samples.dremio.com"."zips.json",
  "context": [
    "@dremio",
    "secondScriptsFolder"
  ],
  "owner": "a97c694f-1e55-4f34-91aa-97f99fee802e"
}'
```

Response

```
{
  "id": "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
  "name": "updatedScript",
  "content": "SELECT * FROM Samples.\"samples.dremio.com\".\"zips.json\"",
  "context": [
    "@dremio",
    "secondScriptsFolder"
  ],
  "owner": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "createdAt": "2024-05-24T17:42:00.304Z",
  "createdBy": "a97c694f-1e55-4f34-91aa-97f99fee802e",
  "modifiedAt": "2024-05-24T18:56:59.409Z",
  "modifiedBy": "a97c694f-1e55-4f34-91aa-97f99fee802e"
```

```
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

Delete a Script

Delete the specified script.

Method and URL

```
DELETE /api/v3/scripts/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the script that you want to delete.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
No response
```

Response Status Codes

204 No Content

401 Unauthorized

Delete a Group of Scripts

Delete the listed group of scripts.

Method and URL

```
POST /api/v3/scripts:batchDelete
```

Parameters

ids Body Array of String

Array of unique identifiers of the scripts that you want to delete.

Example:

```
["74cfddfd-cb0b-4b2f-b555-cb8b827fec1e","37dab994-3f1f-4de7-b2e7-49cb5ff0b395","02fef13e-cedd-46ac-b5bf-abcdcd092146"]
```

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/scripts:batchDelete' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "ids": [
    "74cfddfd-cb0b-4b2f-b555-cb8b827fec1e",
    "37dab994-3f1f-4de7-b2e7-49cb5ff0b395",
    "02fef13e-cedd-46ac-b5bf-abcdcd092146"
  ]
}'
```

Response

```
{
  "unauthorizedIds": [],
  "notFoundIds": [],
  "otherErrorIds": []
}
```

Response Status Codes

200 OK

401 Unauthorized

Retrieve Privilege Information for a Script

Retrieve information about the privileges granted on the specified script.

Method and URL

```
GET /api/v3/scripts/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the script whose privilege information you want to retrieve.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e/grants' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "users": [
    {
      "granteeId": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE",
        "MANAGE_GRANTS"
      ]
    },
    {
      "granteeId": "a97c694f-1e55-4f34-91aa-97f99fee802e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": []
}
```

```
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Update Privileges Granted on a Script

Update the privileges that are granted on the specified script.

Method and URL

```
PUT /api/v3/scripts/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the script whose privilege grants you want to update.

Example: 74cfddfd-cb0b-4b2f-b555-cb8b827fec1e

users Body Array of Object Optional

Array of objects that specify which users should have privileges on the script, as well as each user's specific privileges.

roles Body Array of Object Optional

Array of objects that specify which roles should have privileges on the script, as well as each role's specific privileges.

Attributes of Objects in the `users` Array

granteeld Body String Optional

Unique identifier for the user for whom you want to add or update privileges.

Example: 8be516f3-04c4-4d19-824d-5a70b3c4442e

privileges Body Array of String Optional

The array of privileges you want to add or update for the user.

Enum: VIEW, MODIFY, DELETE, MANAGE_GRANTS

Example: ["VIEW","MODIFY","DELETE"]

Attributes of Objects in the `roles` Array

granteeId Body String Optional

Unique identifier for the role for which you want to add or update privileges.

Example: 6a1725a3-5721-44e3-b64f-0b39a35749ab

privileges Body Array of String Optional

The array of privileges you want to add or update for the role.

Enum: VIEW, MODIFY, DELETE, MANAGE_GRANTS

Example: ["VIEW","MODIFY","DELETE"]

Example

Request

```
curl -X PUT \
'https://{hostname}/api/v3/scripts/74cfddfd-cb0b-4b2f-b555-cb8b827fec1e/grants' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "users": [
    {
      "granteeId": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE"
      ]
    }
  ],
  "roles": [
    {
      "granteeId": "6a1725a3-5721-44e3-b64f-0b39a35749ab",
      "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE"
      ]
    }
  ]
}'
```

Response

```
{
  "users": [
    {
      "granteeId": "8be516f3-04c4-4d19-824d-5a70b3c4442e",
      "privileges": [
```

```

        "VIEW",
        "MODIFY",
        "DELETE"
    ]
},
{
    "granteeId": "a97c694f-1e55-4f34-91aa-97f99fee802e",
    "privileges": [
        "VIEW",
        "MODIFY",
        "DELETE",
        "MANAGE_GRANTS"
    ]
}
],
"roles": [
    {
        "granteeId": "6a1725a3-5721-44e3-b64f-0b39a35749ab",
        "privileges": [
            "VIEW",
            "MODIFY",
            "DELETE"
        ]
    }
]
}
]
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Was this page helpful?

[\[Previous](#)

[Role\] \(/current/reference/api/roles/\) \[Next](#)

[Search\] \(/current/reference/api/search/\)](#)

[Scripts Attributes](#)

[Create a Script](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve All Scripts](#)

Parameters

Example

Response Status Codes

Retrieve a Script by ID

Parameters

Example

Response Status Codes

Update a Script

Parameters

Example

Response Status Codes

Delete a Script

Parameters

Example

Response Status Codes

Delete a Group of Scripts

Parameters

Example

Response Status Codes

Retrieve Privilege Information for a Script

Parameters

Example

Response Status Codes

Update Privileges Granted on a Script

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/search/>

Version: current [26.x]

On this page

Search

Use the Search API to query Dremio entities such as tables, jobs, views, scripts, and Reflections. The API also provides observability into the search infrastructure's health.

Search Object

```
{
  "sessionId": "210be060-b680-43ab-8333-f519847802dd",
  "nextPageToken": "eyJwYWdlVG9rZW4iOiI1MCIsInF1ZXJ5Ijoiam9iIiwizmlsdGVyIjoiIn0",
  "results": [
    {
      "catalogObject": {
        "path": ["Prod", "Sales", "Orders"],
        "branch": "main",
        "type": "TABLE",
        "labels": ["finance", "monthly"],
        "wiki": "Contains monthly sales data",
        "owner": {
          "id": "user-1",
          "type": "USER",
          "username": "data_admin"
        },
        "createdAt": "2023-01-01T10:00:00Z",
        "modifiedAt": "2024-01-01T10:00:00Z",
        "columns": ["order_id", "amount", "order_date"]
      },
    }
  ]
}
```

Search Attributes

sessionId String

Session identifier to correlate API calls during feedback collection.

Example: "210be060-b680-43ab-8333-f519847802dd"

nextPageToken String

Token of the next page of results to fetch in a paginated response.

Example: "eyJwYWdlVG9rZW4iOiI1MCIsInF1ZXJ5Ijoiam9iIiwizmlsdGVyIjoiIn0"

results Array of Object

Array of search results. Each result object can contain a catalogObject, jobObject, scriptObject, or reflectionObject.

Perform a Search

Search for indexed entities such as tables, views, Reflections, jobs, functions, scripts, and folders.

Method and URL

```
POST /api/v3/search
```

Parameters

query Body String

Search string.

Example: "views with date columns"

filter Body String

Optional Common Expression Language (CEL) filter expression to refine the search.

Example: category in ["TABLE"]

pageToken Body String Optional

Token used to retrieve the next page of results.

Example: "eyJwYWdlVG9rZW4iOiI1MCIslInF1ZXJ5Ijoiam9iliwiZmlsdGVyIjoiln0"

maxResults Body Integer Optional

Maximum number of results to return per page.

Example: 10

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/search' \
  --header 'Content-Type: application/json' \
  --header 'Authorization: Bearer <PersonalAccessToken>' \
  --data-raw '{
    "query": "revenue",
    "filter": "category in [\"JOB\"]",
    "pageToken": "",
    "maxResults": 20
  }'
```

Response

```
{
  "sessionId": "6e90f536-d4bd-4ab5-b01e-f81aa4f3002d",
```

```

"results": [
  {
    "category": "job",
    "jobObject": {
      "id": "1802c3f8-05cc-a9fa-417b-0539c5d17700",
      "queriedDatasets": [],
      "sql": " \"SELECT * FROM test\"",
      "jobType": "UI_PREVIEW",
      "user": {
        "id": "788a3438-db1b-483d-a75f-11b6a5d36f31",
        "type": "USER",
        "username": "dremio"
      },
      "startTime": "2025-04-14T16:47:03.981Z",
      "finishTime": "2025-04-14T16:47:03.991Z",
      "jobState": "COMPLETED",
      "error": ""
    }
  },
  {
    "category": "job",
    "jobObject": {
      "id": "1802c419-06fb-be51-9cc2-e158ebaeea00",
      "queriedDatasets": [],
      "sql": "grant all on engine \"test\" to user \"olivier\"",
      "jobType": "UI_PREVIEW",
      "user": {
        "id": "788a3438-db1b-483d-a75f-11b6a5d36f31",
        "type": "USER",
        "username": "dremio"
      },
      "startTime": "2025-04-14T16:46:30.717Z",
      "finishTime": "2025-04-14T16:46:30.730Z",
      "jobState": "FAILED",
      "error": "GRANT on ENGINE is not supported in this edition."
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

500 Internal Server Error

Retrieve Search Infrastructure Status

Method and URL

GET /api/v3/search/status

Retrieves the current state of the search system, including OpenSearch health and indexing status.

Example Response

```
{
  "initializationStatus": {
    "runningTask": "PLAN_WORKFLOW",
    "workFlowJson": "...",
    "upgradeWorkflowState": "Running"
  },
  "openSearchState": {
    "healthStatus": "Yellow",
    "indices": [
      {
        "name": "search-index.1743971124632",
        "docsCount": 4490,
        "size": "70.6mb",
        "health": "green"
      }
    ],
    "aliases": [
      {
        "alias": "search-index-alias",
        "index": "search-index.1743971124632",
        "writeIndex": false
      }
    ]
  }
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

500 Internal Server Error

Remove All Search Documents

Method and URL

POST /api/v3/search/recovery/removeDocuments

Triggers removal of all indexed documents and artifacts. OpenSearch cluster reinitialization is triggered asynchronously.

***Caution:** Use only if index corruption or inconsistency is diagnosed.

Response Status Codes

200 OK

401 Unauthorized

500 Internal Server Error

Example Response

```
{
  "message": "Document removal request has been initiated. You can check the status by calling the /search/status API."
}
```

Object Schemas

searchResultObject Object

Search result object. One of the following fields will be populated based on the result type.

category String

The type of the result object.

Enum: FOLDER, SPACE, TABLE, VIEW, REFLECTION, UDF, SCRIPT, JOB, SOURCE

catalogObject Object Optional

If the result is a catalog object, this field contains the SearchCatalogObject.

Example: { "path": ["testGlue", "folder", "table"], "type": "TABLE", "labels": [], "createdAt": "2024-09-11T17:59:19.162Z", "modifiedAt": "2024-09-11T17:59:19.164Z", "columns": [], "owner": { "id": "ca63ab73-8ee1-467b-811c-224b8f47b9e0", "type": "USER", "username": "user@domain.com" } }

jobObject Object Optional

If the result is a job, this field contains the SearchJobObject.

Example: { "id": "3f6c6332-18bf-487c-b73e-157e91628aaa", "queriedDatasets": [], "sql": "REFRESH DATASET 'source.test-database.test-table'", "jobType": "METADATA_REFRESH", "user": { "id": "2", "type": "USER", "username": "internal" }, "startTime": "2025-04-14T15:26:14.825Z", "finishTime": "2025-04-14T15:26:39.421Z", "jobState": "COMPLETED" }

reflectionObject Object Optional

If the result is a Reflection, this field contains the SearchReflectionObject.

Example: { "id": "31ac58b6-0f2f-4d95-b44b-0f80846a3bd3", "name": "Raw Reflection", "datasetType": "table", "datasetPath": ["Samples", "samples.dremio.com", "NYC-taxi-trips-iceberg"], "createdAt": "2025-04-15T17:28:16.281Z", "modifiedAt": "2025-04-15T17:28:16.281Z", "datasetBranch": "" }

scriptObject Object Optional

If the result is a script, this field contains the SearchScriptObject.

Example: { "id": "1b53bf6c-dd9b-4f16-90e2-23b3c4cfd04f", "name": "test", "owner": { "id": "67f66278-d464-4b7f-b564-e3d58a85bbaf", "type": "USER", "username": "internal" }, "content": "SELECT *\nFROM\nSamples.\"samples.dremio.com\".\"NYC-taxi-trips-iceberg\" w\n\n", "createdAt": "2025-04-14T20:48:29.470Z", "modifiedAt": "2025-04-15T08:14:45.724Z" }

catalogObject Object

Attributes for folders, tables, functions, and views.

path Array of String

Namespace path to the object.

Example: ["Samples","samples.dremio.com"]

branch String Optional

Versioned branch name.

Example: "main"

type String

Enum: FOLDER, SPACE, TABLE, FUNCTION, VIEW.

labels Array of String Optional

User-defined labels.

Example: ["eng","test"]

wiki String Optional

Markdown-formatted documentation or notes.

Example:

```
### Orders Table
```

```
This table contains monthly sales data by region.
```

owner Object

User or role object.

Example: { "id": "788a3438-db1b-483d-a75f-11b6a5d36f31", "type": "USER", "username": "dremio" }

createdAt DateTime

Creation timestamp (RFC 3339).

Example: "2025-04-14T16:47:03.981Z"

modifiedAt DateTime

Last modification timestamp (RFC 3339).

Example: "2025-04-14T16:47:03.981Z"

columns Array of String Optional

Column names.

Example: ["a", "b"]

functionSql String Optional

SQL definition for functions.

Example:

For function created with:

```
-- Example function
CREATE FUNCTION total(a INT, b INT) RETURNS INT RETURN a + b
```

The functionSql is:

```
SELECT a + b
```

jobObject Object

Attributes for jobs.

id String

Unique identifier for the job.

Example: "1802c3f8-05cc-a9fa-417b-0539c5d17700"

queriedDatasets Array of Object

Datasets queried in the job.

Example: [{ "datasetType": "TABLE", "datasetPath": ["Samples", "samples.dremio.com", "NYC-taxi-trips-iceberg"] }]

sql String

Executed SQL statement.

Example: "SELECT * FROM Samples."samples.dremio.com"."NYC-taxi-trips-iceberg"
LIMIT 5"

jobType String

Type of job: UI_RUN, JDBC, METADATA_REFRESH, etc.

Example: "UI_RUN"

user Object

User or role who ran the job.

Example: { "id": "788a3438-db1b-483d-a75f-11b6a5d36f31", "type": "USER",
"username": "dremio" }

startTime DateTime

Job start timestamp.

Example: "2025-04-14T16:47:03.981Z"

finishTime DateTime

Job completion timestamp.

Example: "2025-04-14T16:47:03.981Z"

jobState String

Job status: COMPLETED, FAILED, CANCELED.

Example: "COMPLETED"

error String Optional

Error message if the job failed.

Example: "Object 'Samples.samples.dremio.com.NYC-taxi-trips' not found. Please check
that it exists in the selected context."

scriptObject Object

Attributes for SQL scripts.

id String

Script identifier.

Example: "396eac66-f5d5-4f46-9c51-9d26dd3fc612"

name String

Name of the script.

Example: "test"

owner Object

User or role object.

Example: { "id": "788a3438-db1b-483d-a75f-11b6a5d36f31", "type": "USER", "username": "dremio" }

content String

SQL content.

Example:

```
SELECT *  
FROM Samples."samples.dremio.com"."NYC-taxi-trips.csv"  
LIMIT 1
```

createdAt DateTime

Creation timestamp.

Example: "2025-04-14T16:47:03.981Z"

modifiedAt DateTime

Last modified timestamp.

Example: "2025-04-14T16:47:03.981Z"

reflectionObject Object

Attributes for Reflections.

id String

Reflection ID.

Example: "8711a739-3cc5-4830-8f71-8ea5525291d0"

name String

Name of the Reflection.

Example: "Raw Reflection"

datasetType String

Enum: TABLE, VIEW.

Example: "TABLE"

datasetPath Array of String

Path to the dataset.

Example: ["Samples", "samples.dremio.com", "NYC-taxi-trips-iceberg"]

datasetBranch String

Dataset branch.

Example: "main"

createdAt DateTime

Creation timestamp.

Example: "2025-04-14T16:47:03.981Z"

modifiedAt DateTime

Last modified time.

UserOrRole Object

Used in owner or user fields.

id String

Unique ID of the user or role.

type String

Enum: USER, ROLE.

username String Conditional

Present for USER.

Example: "data_admin"

roleName String Conditional

Present for ROLE.

Example: "ADMIN"

Was this page helpful?

[Previous

Scripts](/current/reference/api/scripts/)[Next

Source](/current/reference/api/source)

Search Attributes

Perform a Search

Parameters

Example

Response Status Codes

Retrieve Search Infrastructure Status

Example Response

Response Status Codes

Remove All Search Documents

Response Status Codes

Example Response

Object Schemas

searchResultObject Object

catalogObject Object

jobObject Object

scriptObject Object

reflectionObject Object

UserOrRole Object

Source:
<https://docs.dremio.com/current/reference/api/s>
ource

Version: current [26.x]

On this page

Source

Use the Source API to clear the AWS Lake Formation permission cache for AWS Glue Data Catalog sources.

Dremio keeps a cache of permissions defined in AWS Lake Formation with a one-hour expiry time. When the cache for the queried table expires, Dremio requests permission information from AWS Lake Formation. After changing permissions on the AWS Lake Formation side, use the Source API to immediately invalidate Dremio's AWS Lake Formation permission cache.

note

The Source API is supported only for AWS Glue Data Catalog sources.

Clear the Permission Cache

Clear the AWS Lake Formation permission cache for an AWS Glue Data Catalog source.

Method and URL

```
DELETE /api/v3/source/{source-name}/permission-cache
```


Parameters

source-name Path String

The name of the AWS Glue Data Catalog source whose Lake Formation permission cache you want to clear.

Example: glueProd

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/source/glueProd/permission-cache' \  
--header 'Authorization: Bearer <personal access token>' \  
--header 'Content-Type: application/json'
```

Response

No response

Responses

204 No Content

400 Bad Request

404 Not Found

415 Unsupported Media Type

500 Internal Server Error

Was this page helpful?

[Previous

Search]([current/reference/api/search/])[Next

SQL]([current/reference/api/sql/])

Clear the Permission Cache

Parameters

Example

Responses

Source:
[https://docs.dremio.com/current/reference/api/s
ql/](https://docs.dremio.com/current/reference/api/sql/)

Version: current [26.x]

On this page

SQL

Use the SQL API to submit SQL queries. The response contains the ID for the job associated with the SQL query. Use the job ID in [Job API](#) requests to get more information about the job, including results.

Submit an SQL Query

Submit an SQL query and retrieve the associated job ID for use in [Job API](#) requests.

Method and URL

```
POST /api/v3/sql
```

Parameters

sql Body String

SQL query to run.

note

Double-quotation marks within a SQL statement need to be escaped.

Example: `SELECT * FROM Samples."samples.dremio.com"."SF weather 2018-2019.csv"`

context Body Array of String Optional

Path to the container where the query should run within Dremio, expressed as an array. The path consists of the source or space, followed by the folder and subfolders.

Example: `["Samples","samples.dremio.com"]`

[references](#) Body Object Optional

References to the specific versions (branches, tags, and commits) in Nessie sources where you want to run the SQL query. If references are not specified for a Nessie source, the SQL query runs on the default branch.

Example: `{"nessieSource1": {"type": "BRANCH","value": "testing"},"nessieSource2": {"type": "TAG","value": "Test commit"},"nessieSource3": {"type": "COMMIT","value": "7a5edb57e035f52beccfab632cea070514eb8b773f616aaeaf668e2f0be8f10d"}}`

Parameters of the `references` Object

[<Nessie source>](#) Body String Optional

The name of the Nessie source where you want to run the SQL query.

Example: nessieSource1

Parameters of the `<Nessie source>` Object

type Body String Optional

The type of Nessie source object where you want to run the SQL query.

Enum: BRANCH, TAG, COMMIT

Example: BRANCH

value Body String Optional

The branch or tag name or commit hash in the Nessie source on which you want to run the SQL query.

Example: testing

Example Request Using Only the SQL Parameter

```
curl -X POST 'https://{hostname}/api/v3/sql' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "sql": "SELECT * FROM Samples.\"samples.dremio.com\".\"SF weather 2018-2019.csv\""
}'
```

Example Request Using Optional Parameters

```
curl -X POST 'https://{hostname}/api/v3/sql' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "sql": "SELECT * FROM \"SF weather 2018-2019.csv\"",
  "context": [
    "Samples",
    "samples.dremio.com"
  ],
  "references": {
    "nessieSource1": {
      "type": "BRANCH",
      "value": "testing"
    },
    "nessieSource2": {
      "type": "TAG",
      "value": "Test commit"
    },
    "nessieSource3": {
      "type": "COMMIT",
      "value": "7a5edb57e035f52beccfab632cea070514eb8b773f616aaeaf668e2f0be8f10d"
    }
  }
}'
```

Example Response

```
{  
  "id": "2f067496-7cf0-a70e-0222-34d53a5dc800"  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

404 Not Found

Was this page helpful?

[Previous

Source](/current/reference/api/source)[Next

User](/current/reference/api/user/)

Submit an SQL Query

Parameters

Example Response

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/user/>

Version: current [26.x]

On this page

User Enterprise

The User API manages local and SSO users and service users.

Users

Service Users

User Object

```
{  
  "@type": "EnterpriseUser",  
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",  
  "name": "dremio",  
}
```

```

"firstName": "Dre",
"lastName": "Mio",
"email": "user@dremio.com",
"tag": "EuCNt1nnvdI=",
"roles": [
  {
    "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
    "name": "PUBLIC",
    "type": "SYSTEM"
  },
  {
    "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
    "name": "ADMIN",
    "type": "SYSTEM"
  }
],
"source": "local",
"identityType": "REGULAR_USER",
"active": true
}

```

Service User Object

```

{
  "@type": "EnterpriseUser",
  "id": "5263eb65-5fba-406b-9539-8627240adb8e",
  "name": "service_user",
  "roles": [
    {
      "id": "1934dbb5-c57b-49fa-b7da-e709c1716c2f",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "563fa12c-fa11-4753-9025-f1089ab8f1bc",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ],
  "source": "local",
  "active": true,
  "identityType": "SERVICE_USER",
  "oauthClientId": "61d00e34-a938-4b80-9a98-063fc723bd6d"
}

```

Attributes

@type String

Type of user. Valid values are:

-

`EnterpriseUser`: Created by Dremio Enterprise Edition.

`User`: Created by Dremio Community Edition.

Example: EnterpriseUser

id String (UUID)

Unique identifier of the user.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

name String

Username of the Dremio user account.

Example: dremio

firstName String

User's first name. Available only with identityType `REGULAR_USER`.

Example: Dre

lastName String

User's last name. Available only with identityType `REGULAR_USER`.

Example: Mio

email String

User's email address. If the user is managed with the Dremio Okta application, email is the primary email address in the user's Okta profile. If the user is managed with Microsoft Entra ID, email is the work email address in the user's Microsoft Entra ID profile. Available only with identityType `REGULAR_USER`.

Example: user@dremio.com

tag String

Unique identifier of the user version. Dremio changes the tag whenever the user changes and uses the tag to ensure that PUT requests apply to the most recent version of the user. Available only with identityType `REGULAR_USER`.

Example: EuCNt1nnvdl=

roles Array of Object

Information about the local and referenced external roles to which the user belongs.

Example: [{"id": "8ac1bbca-479c-4c47-87e9-7f946f665c13", "name": "PUBLIC", "type": "SYSTEM"}, {"id": "43dce6d7-40ff-4afa-9901-71c30eb92744", "name": "ADMIN", "type": "SYSTEM"}]

source String

Information about how the user was created.

-

``external``: User was imported with an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

``local``: User was created manually in the Dremio user interface (UI) or with the User API.

Example: local

active Boolean

If the user account is active in Dremio, the value is ``true``. Otherwise, the value is ``false``. The active value is set to ``true`` when the user is created and only changes if the user's status changes in external System for Cross-domain Identity Management (SCIM) provisioning. When the user is activated in the SCIM application, Dremio sets the value to ``true``. When the user is deactivated in the SCIM application, Dremio sets the value to ``false``.

Example: true

identityType String

The type of user:

``REGULAR_USER`` (Default): An individual person.

``SERVICE_USER``: A machine or application user.

Example: REGULAR_USER

oauthClientId String

The unique OAuth client identifier generated by Dremio for service user authentication. This client ID is used in OAuth client credentials flows to obtain access tokens for API authentication. Available only for identityType ``SERVICE_USER``.

Example: 88d7c3cc-cab5-4ab1-917c-e075265501dd

Attributes of Objects in the ``roles`` Array

id String (UUID)

Unique identifier of the role.

Example: 43dce6d7-40ff-4afa-9901-71c30eb92744

name String

Name of the role.

Example: ADMIN

type String

Origin of the role.

``INTERNAL``: Role was created in the Dremio user interface (UI) or with the Role API.

``EXTERNAL``: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

``SYSTEM``: Role was predefined in Dremio.

Example: SYSTEM

List Users

Method and URL

```
GET /api/v3/user
```

Parameters

filter Query Object Optional

Filter the user list:

User Fields	Functions	Operators
* `id` * `name` * `firstName` * `lastName` * `email` * `identityType`	* `startsWith()`	
* `contains()` | * `==` encoded as `%3D%3D` (equals) * `&&` encoded as `%26%26`
(logical and) * `||` encoded as `%7C%7C` (logical or) |

The filter parameter value must be URL-encoded. Examples:

``?filter=name%3D%3D%27john%2Edoe%27`` evaluates to "name=='john.doe'"

``?filter=email%2EstartsWith%28%27admin%27%29`` evaluates to "email.startsWith('admin')"

``?filter=firstName%2Econtains%28%27smith%27%29%26%26identityType%3D%3D%27SERVICE_USER%27`` evaluates to "firstName.contains('smith')&&identityType=='SERVICE_USER'"

For more information, see [filter Query Parameter](#).

Example

Request

```
curl -X GET "https://{hostname}/api/v3/user?filter=firstName%3D%3D%27dre%27%26%26identityType%3D%3D%27REGULAR_USER%27" \
  --header 'Authorization: Bearer C4bVdq0MSFyHW5etR9WvrQP9WNYdRZd0IePtfw0mRJLUU4cWRQWMEZYbx5RDaA==' \
  --header 'Content-Type: application/json'
```

Response


```
{
  "data": [
    {
      "@type": "EnterpriseUser",
      "id": "b5537771-8506-4c8d-8014-6687cf570410",
      "name": "dremio",
      "firstName": "Dre",
      "lastName": "Mio",
      "email": "dremio@dremio.test",
      "tag": "zF+ueKESvHU=",
      "roles": [
        {
          "id": "bfc34dbc-f659-4e84-9278-31f7d9375656",
          "name": "PUBLIC",
          "type": "SYSTEM"
        },
        {
          "id": "90fd20a2-9352-4386-a963-fdee15fb5243",
          "name": "ADMIN",
          "type": "SYSTEM"
        }
      ],
      "source": "local",
      "active": true,
      "identityType": "REGULAR_USER"
    }
  ],
  "totalResults": 1
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

Create a User

Method and URL

POST /api/v3/user

Parameters

name Body String

Username for the Dremio user account. The name must be unique and cannot be updated after the user is created.

Example: dremio

firstName Body String Optional

User's first name. Available only for identityType `REGULAR_USER`.

Example: Dre

lastName Body String Optional

User's last name. Available only for identityType `REGULAR_USER`.

Example: Mio

email Body String Optional

User's email address. Available only for identityType `REGULAR_USER`.

Example: user@dremio.com

roles Body Array of Object Optional

Information about the roles to which the user should be assigned. All users are assigned to the PUBLIC role by default.

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role.

Example: 43dce6d7-40ff-4afa-9901-71c30eb92744

name Body String

Name of the role. All users are assigned to the PUBLIC role by default.

Example: ADMIN

type Body String Optional

Origin of the role.

`INTERNAL`: Role was created in the Dremio user interface (UI) or with the Role API.

`EXTERNAL`: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

`SYSTEM`: Role was predefined in Dremio.

Example: SYSTEM

Example

Create a User

Create a Service User

Request

```
curl -X POST 'https://{hostname}/api/v3/user' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ]
}'
```

Response

```
{
  "@type": "EnterpriseUser",
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "firstName": "Dre",
  "lastName": "Mio",
  "email": "user@dremio.com",
  "tag": "EuCNt1nnvdI=",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ],
  "source": "external",
  "identityType": "REGULAR_USER",
  "active": true
}
```

Request

```
curl -X POST 'https://{hostname}/api/v3/user' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json' \  
--data-raw '{  
  "name": "api_service_user",  
  "description": "Nightly data ingestion",  
  "roles": [  
    {  
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",  
      "name": "PUBLIC",  
      "type": "SYSTEM"  
    },  
    {  
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",  
      "name": "ADMIN",  
      "type": "SYSTEM"  
    }  
  ],  
  "identityType": "SERVICE_USER"  
}
```

Response

```
{  
  "@type": "EnterpriseUser",  
  "id": "090d605c-ba0e-4b42-8831-e10d6fd08101",  
  "name": "api_service_user",  
  "roles": [  
    {  
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",  
      "name": "PUBLIC",  
      "type": "SYSTEM"  
    },  
    {  
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",  
      "name": "ADMIN",  
      "type": "SYSTEM"  
    }  
  ],  
  "source": "local",  
  "active": true,  
  "identityType": "SERVICE_USER",  
  "oauthClientId": "3f7a9ff7-fd5f-493c-b4f0-8d1a76ceef24"  
}
```

Response Status Codes

200 OK

204 No Content

400 Bad Request
401 Unauthorized
404 Not Found
405 Method Not Allowed
500 Internal Server Error

Retrieve a User by ID

Retrieve a specific user by the user's ID.

Method and URL

```
GET /api/v3/user/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user you want to retrieve.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
{  
  "@type": "EnterpriseUser",  
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",  
  "name": "dremio",  
  "firstName": "Dre",  
  "lastName": "Mio",  
  "email": "user@dremio.com",  
  "tag": "EuCNT1nnvdI=",  
  "roles": [  
    {  
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",  
      "name": "PUBLIC",  
      "type": "SYSTEM"  
    },  
    {  
      "id": "43dce6d7-40ff-4afa-9901-71c30eb92744",
```

```
    "name": "ADMIN",
    "type": "SYSTEM"
  },
  "source": "local",
  "identityType": "REGULAR_USER",
  "active": true
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Retrieve a User by Name

Use this command to import a user from your integrated identity provider (IdP) by specifying their name.

Method and URL

```
GET /api/v3/user/by-name/{name}
```

Parameters

name Path String

Username of the user you want to retrieve. Usernames are case-insensitive. If the username includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: nightly-etl

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/user/by-name/nightly-etl' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "@type": "EnterpriseUser",
  "id": "5263eb65-5fba-406b-9539-8627240adb8e",
  "name": "service_user",
  "roles": [
    {
      "id": "1934dbb5-c57b-49fa-b7da-e709c1716c2f",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "563fa12c-fa11-4753-9025-f1089ab8f1bc",
      "name": "ADMIN",
      "type": "SYSTEM"
    }
  ],
  "source": "local",
  "active": true,
  "identityType": "SERVICE_USER",
  "oauthClientId": "61d00e34-a938-4b80-9a98-063fc723bd6d"
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

500 Internal Server Error

Update a User

Update the specified user.

Method and URL

PUT /api/v3/user/{id}

Parameters

id Path String (UUID)

Unique identifier of the user you want to update.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

id Body String (UUID)

Unique identifier of the user you want to update.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag Body String

Unique identifier of the user version to update. Dremio uses the tag to ensure that you are updating the most recent version of the user. Available only with identityType `REGULAR_USER`.

Example: BNGRmgfEnDg=

name Body String

Name of the user. Available only with identityType `REGULAR_USER`.

Example: dremio

firstName Body String Optional

User's first name. Available only with identityType `REGULAR_USER`.

Example: Dre

lastName Body String Optional

User's last name. Available only with identityType `REGULAR_USER`.

Example: Mio

email Body String Optional

User's email address. Available only with identityType `REGULAR_USER`.

Example: user@dremio.com

roles Body Array of Object Optional

Information about the roles to which the user should be assigned. All users are assigned to the PUBLIC role by default.

Example: [{"id": "8ac1bbca-479c-4c47-87e9-7f946f665c13", "name": "PUBLIC", "type": "SYSTEM"}, {"id": "43dce6d7-40ff-4afa-9901-71c30eb92744", "name": "ADMIN", "type": "SYSTEM"}]

Parameters of Objects in the `roles` Array

id Body String (UUID)

Unique identifier of the role.

Example: 43dce6d7-40ff-4afa-9901-71c30eb92744

name Body String

Name of the role. All users are assigned to the PUBLIC role by default.

Example: VIEWER

type Body String Optional

Origin of the role.

`INTERNAL`: Role was created in the Dremio user interface (UI) or with the Role API.

`EXTERNAL`: Role was imported from an external service like Microsoft Entra ID, Lightweight Directory Access Protocol (LDAP), or a System for Cross-domain Identity Management (SCIM) provider.

`SYSTEM`: Role was predefined in Dremio.

Example: INTERNAL

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "tag": "EuCNt1nnvdI=",
  "firstName": "Dremio",
  "lastName": "User",
  "email": "user@dremio.com",
  "roles": [
    {
      "id": "2f498015-9211-4b15-8fc0-493628ae7b6e",
      "name": "VIEWER"
    }
  ]
}'
```

Response

```
{
  "@type": "EnterpriseUser",
  "id": "b9dbebc7-bc3b-4d56-9154-31762ab65a43",
  "name": "dremio",
  "firstName": "Dremio",
  "lastName": "User",
  "email": "user@dremio.com",
  "tag": "BE1LYg3cmAk=",
  "roles": [
    {
      "id": "8ac1bbca-479c-4c47-87e9-7f946f665c13",
      "name": "PUBLIC",
      "type": "SYSTEM"
    },
    {
      "id": "2f498015-9211-4b15-8fc0-493628ae7b6e",
      "name": "VIEWER",
      "type": "INTERNAL"
    }
  ]
}
```

```
}  
],  
"source": "external",  
"identityType": "REGULAR_USER",  
"active": true  
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Delete a User

Method and URL

```
DELETE /api/v3/user/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user you want to delete. You can only delete users that are not currently logged in to Dremio.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

version Query String

When the identityType is `REGULAR_USER`, the version to delete. The version value is the user's tag, which you can find in the response for a request to [Retrieve a User by ID](#) or [Retrieve a User by Name](#). Dremio uses the version value to ensure that you are deleting the most recent version of the user. If you provide an incorrect tag, the response includes an error message that lists the correct tag for the specified user ID.

Example: ?version=BE1LYg3cmAk=

Example

Delete a User

Delete a Service User

Request

```
curl -X DELETE 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43?version=BE1LYg3cmAk=' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

A successful request returns an empty response with the HTTP `204 No Content` status.

Request

```
curl -X DELETE 'https://{hostname}/api/v3/user/5263eb65-5fba-406b-9539-8627240adb8e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

A successful request returns an empty response with the HTTP `204 No Content` status.

Response Status Codes

204 No Content

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

SQL](/current/reference/api/sql/)[Next

Credentials](/current/reference/api/user/credentials)

Attributes

List Users

Parameters

Example

Response Status Codes

Create a User

Parameters

Example

Response Status Codes

Retrieve a User by ID

Parameters

Example

Response Status Codes

Retrieve a User by Name

Parameters

Example

Response Status Codes

Update a User

Parameters

Example

Response Status Codes

Delete a User

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/wlm/>

Version: current [26.x]

Workload Management Enterprise

Use the Workload Management API to manage cluster resources and workloads by defining queues that have specific characteristics like memory limits, CPU priority, and queueing and runtime timeouts. You can also define rules that determine how to assign queries to specific queues.

See [Workload Management](#) for more information about Dremio's workload-management features.

Was this page helpful?

[Previous

User](/current/reference/api/user/)[Next

Queue](/current/reference/api/wlm/queue)

Source:
<https://docs.dremio.com/current/reference/api/reflections/reflection-summary/>

Version: current [26.x]

On this page

Reflection Summary Enterprise

Use the Reflection API to retrieve a Reflection summary that includes the raw and aggregation Reflections for the Dremio instance.

Reflection summary objects are different from Reflection objects. Reflection summaries do not include certain attributes that define the Reflection, like the display, dimension, measure, sort, and partition attributes. Reflection summaries do include several attributes that do not appear in Reflection objects, like `datasetType`, `datasetPath`, and counts and links for considered, matched, and chosen jobs.

Reflection Summary Object

```
{
  "data": [
    {
      "createdAt": "2022-07-05T19:19:40.244Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
      "totalSizeBytes": 9272,
      "datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
      "datasetType": "VIRTUAL_DATASET",
      "datasetPath": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "status": {
        "configStatus": "OK",
        "refreshStatus": "MANUAL",
        "availabilityStatus": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "refreshMethod": "FULL",
        "failureCount": 0,
        "lastDataFetchAt": "2023-01-13T19:05:03.532Z",
        "expiresAt": "3022-05-16T19:46:02.342Z",
        "lastRefreshDurationMillis": 46387
      },
      "consideredCount": 202,
      "matchedCount": 45,
      "chosenCount": 5,
      "consideredJobsLink":
```

```

"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-26T23:27:04.281Z",
    "updatedAt": "2023-01-26T23:27:04.281Z",
    "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
    "reflectionType": "RAW",
    "name": "Raw Reflection",
    "currentSizeBytes": 0,
    "outputRecords": -1,
    "totalSizeBytes": 0,
    "datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
    "datasetType": "PHYSICAL_DATASET",
    "datasetPath": [
      "@dremio",
      "test"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "GIVEN_UP",
      "availabilityStatus": "NONE",
      "combinedStatus": "FAILED",
      "refreshMethod": "NONE",
      "failureCount": 3,
      "lastFailureMessage": "The Default engine is not online.",
      "lastDataFetchAt": null,
      "expiresAt": null,
      "lastRefreshDurationMillis": -1
    },
    "consideredCount": 0,
    "matchedCount": 0,
    "chosenCount": 0,
                                                                 "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
                                                                 "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,

```

```

    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:19.030Z",
    "updatedAt": "2023-01-13T19:50:19.030Z",
    "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
    "reflectionType": "AGGREGATION",
    "name": "dataset991_agg991",
    "currentSizeBytes": 9273,
    "outputRecords": 51,
    "totalSizeBytes": 9273,
    "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
    "datasetType": "VIRTUAL_DATASET",
    "datasetPath": [
      "temp",
      "dataset991"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "MANUAL",
      "availabilityStatus": "AVAILABLE",
      "combinedStatus": "CAN_ACCELERATE",
      "refreshMethod": "FULL",
      "failureCount": 0,
      "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
      "expiresAt": "3022-05-16T19:46:02.342Z",
      "lastRefreshDurationMillis": 11697
    },
    "consideredCount": 60,
    "matchedCount": 9,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:17.714Z",
    "updatedAt": "2023-01-13T19:50:17.714Z",
    "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
    "reflectionType": "RAW",
    "name": "dataset991_raw991",
    "currentSizeBytes": 818790,
    "outputRecords": 29467,

```

```

"totalSizeBytes": 818790,
"datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
"datasetType": "VIRTUAL_DATASET",
"datasetPath": [
  "temp",
  "dataset991"
],
"status": {
  "configStatus": "OK",
  "refreshStatus": "MANUAL",
  "availabilityStatus": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "refreshMethod": "FULL",
  "failureCount": 0,
  "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
  "expiresAt": "3022-05-16T19:46:02.747Z",
  "lastRefreshDurationMillis": 16666
},
"consideredCount": 54,
"matchedCount": 37,
"chosenCount": 0,
"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
}
],
"nextPageToken": "CiQxNjc0MjhlYi03OTM2LTRlYTYtYTFmYi0yM2IxYWY2ZTk0NTQSA==",
"isCanAlterReflections": true
}

```

Reflection Summary Attributes

data Array of Object

List of Reflection-summary objects for each Reflection in the Dremio instance.

nextPageToken String

Opaque string to pass for the `pageToken` query parameter in the next request to retrieve the next set of results. If nextPageToken is not included in the response, all available resources have been returned.

Example: CiQxNjc0MjhIYi03OTM2LTTRIYtYTFmYi0yM2IxYWM2ZTk0NTQSAA==

isCanAlterReflections Boolean

If the current user has project-level privileges to alter Reflections, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of Objects in the `data` Array

createdAt String

Date and time that the Reflection was created, in UTC format.

Example: 2022-07-05T19:19:40.244Z

updatedAt String

Date and time that the Reflection was last updated, in UTC format.

Example: 2023-01-13T19:46:01.313Z

id String (UUID)

Unique identifier of the Reflection.

Example: 27077c03-ae49-454c-a7bb-a9a8b5eca224

reflectionType String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio UI, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: NYC_taxi_agg

currentSizeBytes Integer

Data size of the latest Reflection job (if one exists), in bytes.

Example: 9272

outputRecords Integer

Number of records returned for the latest Reflection.

Example: 51

totalSizeBytes Integer

Data size of all Reflection jobs that have not been pruned (if any exist), in bytes.

Example: 9272

datasetId String

Unique identifier of the anchor dataset that is associated with the Reflection.

Example: fa7c487f-9550-474e-8a41-4826564c6b09

datasetType String

Type for the anchor dataset that is associated with the Reflection. If the anchor dataset is a table, the type is `PHYSICAL_DATASET`. If the anchor dataset is a view, the type is `VIRTUAL_DATASET`.

Enum: PHYSICAL_DATASET, VIRTUAL_DATASET

Example: VIRTUAL_DATASET

datasetPath String

Path to the anchor dataset that is associated with the Reflection within Dremio, expressed in an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the dataset itself as the last item in the array.

Example: ["Samples","samples.dremio.com","NYC Taxi Trips"]

status Object

Information about the status of the Reflection.

Example: {\n "configStatus": "OK",\n "refreshStatus": "MANUAL",\n "availabilityStatus": "AVAILABLE",\n "combinedStatus": "CAN_ACCELERATE",\n "refreshMethod": "FULL",\n "failureCount": 0,\n "lastDataFetchAt": "2023-01-13T19:05:03.532Z",\n "expiresAt": "3022-05-16T19:46:02.342Z",\n "lastRefreshDurationMillis": 46387\n }

consideredCount Integer

Number of jobs that considered the Reflection during planning.

Example: 202

matchedCount Integer

Number of jobs that matched the Reflection during planning.

Example: 45

chosenCount Integer

Number of jobs accelerated by the Reflection.

Example: 5

consideredJobsLink String

Link to list of considered jobs for the Reflection.

Example:

```
/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D
```

matchedJobsLink String

Link to list of matched jobs for the Reflection.

Example:

```
/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D
```

chosenJobsLink String

Link to list of chosen jobs for the Reflection.

Example:

```
/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D
```

isArrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, the value is `true`. Otherwise, the value is `false`.

Example: false

isCanView Boolean

If you can view Reflections on all datasets of a source, system, space, or folder, the value is `true`. Otherwise, the value is `false`.

Example: true

isCanAlter Boolean

If you can create, edit, and view Reflections on all datasets of a source, system, space, or folder, the value is `true`. Otherwise, the value is `false`.

Example: true

isEnabled Boolean

If the Reflection is available for accelerating queries, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of the `status` Object

configStatus String

Status of the Reflection configuration. If the value is `OK`, the Reflection configuration

is free of errors. If the value is ``INVALID``, the Reflection configuration contains one or more errors.

Enum: OK, INVALID

Example: OK

refreshStatus String

Status of the Reflection refresh.

``GIVEN_UP``: Dremio attempted to refresh the Reflection multiple times, but each attempt has failed and Dremio will not make further attempts.

``MANUAL``: Refresh period is set to 0, so you must use the Dremio UI to manually refresh the Reflection.

``RUNNING``: Dremio is currently refreshing the Reflection.

``SCHEDULED``: The Reflection refreshes according to a schedule.

``ON_DATA_CHANGES``: All of the Reflection's underlying tables are in Iceberg format, and the Reflection refreshes automatically if new snapshots are created after an update to the underlying tables.

Enum: GIVEN_UP, MANUAL, RUNNING, SCHEDULED, ON_DATA_CHANGES

Example: MANUAL

availabilityStatus String

Status of the Reflection's availability for accelerating queries.

Enum: NONE, EXPIRED, AVAILABLE

Example: AVAILABLE

combinedStatus String

Status of the Reflection based on a combination of config, refresh, and availability.

``CAN_ACCELERATE``: The Reflection is fully functional.

``CAN_ACCELERATE_WITH_FAILURES``: The most recent refresh failed to obtain a status, but Dremio still has a valid materialization.

``CANNOT_ACCELERATE_INITIALIZING``: The Reflection is currently being loaded into the materialization cache. During this time, the Reflection is unable to accelerate queries.

``CANNOT_ACCELERATE_MANUAL``: The Reflection is unable to accelerate any queries, and the ``Never Refresh`` option is selected for the refresh policy.

``CANNOT_ACCELERATE_SCHEDULED``: The Reflection is currently unable to accelerate any queries, but it has been scheduled for a refresh at a future time.

``DISABLED``: The Reflection has been manually disabled.

``EXPIRED``: The Reflection has expired and cannot be used.

``FAILED``: The attempt to refresh the Reflection has failed, typically three times in a row. The Reflection is still usable.

``INVALID``: The Reflection is invalid because the underlying dataset has changed.

``REFRESHING``: The Reflection is currently being refreshed.

Example: `CAN_ACCELERATE`

refreshMethod String

The method used for the most recent refresh of the Reflection. For new Reflections, the value is ``NONE`` until planned. For more information, read [Refreshing Reflections](#).

Enum: `NONE`, `FULL`, `INCREMENTAL`

Example: `FULL`

failureCount Integer

Number of times that an attempt to refresh the Reflection failed.

Example: `0`

lastFailureMessage String

The error message from the last failed Reflection refresh. If the refresh of a Reflection never fails or succeeds after a failure, this attribute does not appear.

Example: `"The Default engine is not online."`

lastDataFetchAt String

Date and time that the Reflection data was last refreshed, in UTC format. If the Reflection is running, failing, or disabled, the `lastDataFetchAt` value is ``1969-12-31T23:59:59.999Z``.

Example: `2023-01-13T19:05:03.532Z`

expiresAt String

Date and time that the Reflection expires, in UTC format. If the Reflection is running, failing, or disabled, the `expiresAt` value is ``1969-12-31T23:59:59.999Z``.

Example: `3022-05-16T19:46:02.342Z`

lastRefreshDurationMillis Integer

Duration of the most recent refresh for the Reflection. In milliseconds.

Example: `46387`

Retrieve a Reflection Summary

Retrieve a summary of the raw and aggregation Reflections in the Dremio instance.

Method and URL

Parameters

pageToken Query String Optional

Token for retrieving the next page of Reflection summary results. If the Dremio instance has more Reflection summary results than the maximum per page (default 50), the response includes a nextPageToken after the data array. Use the nextPageToken value in your request URL as the pageToken value. Do not change any other query parameters included in the request URL when you use pageToken. For more information, read [pageToken Query Parameter](#).

maxResults Query Integer Optional

Maximum number of Reflection summaries to return in the response. Maximum valid value is `100`. Default is `50`. For more information, read [maxResults Query Parameter](#).

filter Query Object Optional

Filters for Reflection name, dataset name, availability status, and refresh status. Value is a URL-encoded string that represents a JSON object. The JSON object specifies the attributes to filter on and the values to match for each attribute. Available filter attributes:

reflectionType: `RAW`, `AGGREGATION` (array of string)

refreshStatus: `GIVEN_UP`, `MANUAL`, `RUNNING`, `SCHEDULED`, `ON_DATA_CHANGES` (array of string)

availabilityStatus: `NONE`, `EXPIRED`, `AVAILABLE` (array of string)

configStatus: `OK`, `INVALID` (array of string)

enabledFlag: `true`, `false` (Boolean)

reflectionNameOrDatasetPath: full or partial Reflection name or dataset path; case insensitive (string)

reflectionIds: IDs of Reflections to retrieve (array of string); must be used alone, with no other filters or query parameters

For more information, read [filter Query Parameter](#).

orderBy Query String Optional

Organize the response in ascending (default) or descending order by reflectionName, datasetName, or reflectionType. To specify descending order, precede the orderBy value with a `-` character. For more information, read [orderBy Query Parameter](#).

Example Request Without Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary'
--header 'Authorization: Bearer <dremioAccessToken>' \
```

```
--header "Content-Type: application/json"
```

Example Response

```
{
  "data": [
    {
      "createdAt": "2023-01-13T19:46:01.313Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
      "totalSizeBytes": 9272,
      "datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
      "datasetType": "VIRTUAL_DATASET",
      "datasetPath": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "status": {
        "configStatus": "OK",
        "refreshStatus": "MANUAL",
        "availabilityStatus": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "refreshMethod": "FULL",
        "failureCount": 0,
        "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
        "expiresAt": "3022-05-16T19:46:02.342Z",
        "lastRefreshDurationMillis": 46387
      },
      "consideredCount": 202,
      "matchedCount": 45,
      "chosenCount": 5,
      "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "isArrowCachingEnabled": false,
      "isCanView": true,
      "isCanAlter": true,
      "isEnabled": true
    },
    {
      "createdAt": "2023-01-26T23:27:04.281Z",
```

```

    "updatedAt": "2023-01-26T23:27:04.281Z",
    "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
    "reflectionType": "RAW",
    "name": "Raw Reflection",
    "currentSizeBytes": 0,
    "outputRecords": -1,
    "totalSizeBytes": 0,
    "datasetId": "acdad4be-7049-47e4-b616-b471c5b3c60c",
    "datasetType": "PHYSICAL_DATASET",
    "datasetPath": [
      "@dremio",
      "test"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "GIVEN_UP",
      "availabilityStatus": "NONE",
      "combinedStatus": "FAILED",
      "refreshMethod": "NONE",
      "failureCount": 3,
      "lastDataFetchAt": null,
      "expiresAt": null,
      "lastRefreshDurationMillis": -1
    },
    "consideredCount": 0,
    "matchedCount": 0,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:19.030Z",
    "updatedAt": "2023-01-13T19:50:19.030Z",
    "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
    "reflectionType": "AGGREGATION",
    "name": "dataset991_agg991",
    "currentSizeBytes": 9273,
    "outputRecords": 51,
    "totalSizeBytes": 9273,
    "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
    "datasetType": "VIRTUAL_DATASET",
    "datasetPath": [
      "temp",

```



```

    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
    "expiresAt": "3022-05-16T19:46:02.342Z",
    "lastRefreshDurationMillis": 11697
  },
  "consideredCount": 60,
  "matchedCount": 9,
  "chosenCount": 0,

  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:17.714Z",
  "updatedAt": "2023-01-13T19:50:17.714Z",
  "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
  "reflectionType": "RAW",
  "name": "dataset991_raw991",
  "currentSizeBytes": 818790,
  "outputRecords": 29467,
  "totalSizeBytes": 818790,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",
    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
    "expiresAt": "3022-05-16T19:46:02.747Z",

```

```

    "lastRefreshDurationMillis": 16666
  },
  "consideredCount": 54,
  "matchedCount": 37,
  "chosenCount": 0,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
}
],
"nextPageToken": "CiQxNjc0MjhlYi030TM2LTRlYtYTFmYi0yM2IxYWM2ZTk0NTQSA==",
"isCanAlterReflections": true
}

```

This endpoint supports query parameters that you can add to the request URL to include only specific types of Reflections in the Reflection summary, specify the maximum number of results to return, and sort the response to list Reflections in ascending or descending order.

For example, to order the Reflections within the summary in ascending order by reflectionName, add `?orderBy=reflectionName` to the request URL. For descending order, add a `-` character before the attribute name: `?orderBy=-reflectionName`.

In the same request, you can add the `filter` query parameter to retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths. The JSON object for such a filter would look like this:

Example JSON Object for Filter

```

{
  "reflectionType": ["RAW"],
  "refreshStatus": ["MANUAL", "SCHEDULED"],
  "enabledFlag": true,
  "reflectionNameOrDatasetPath": "samples.dremio.com"
}

```

However, to use the JSON object in the request URL, you must convert it to URL-encoded JSON, which looks like this:

Example JSON Object in URL-Encoded JSON

```

%7B%0A%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%

```

```
20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D
```

Here is an example request URL that includes both the `orderBy` and `filter` query parameters:

Example Request with orderBy and filter Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName&filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

For this request, the Reflection summary in the response will include only raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths, and the Reflections will be listed in ascending order by reflectionName.

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Recommendations](/current/reference/api/reflections/reflection-recommendations)

Reflection Summary Attributes

Retrieve a Reflection Summary

Parameters

Example Response

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/source/>

Version: current [26.x]

On this page

Source

Use the Catalog API to retrieve information about [sources](#) and the child objects they contain, as well as to create, update, and delete sources.

note

Dremio supports a number of different source types. Each source type has the same parameters *except* for the parameters within the `config` object. The available parameters in the `config` object are different for each source type. The examples on this page use an Amazon S3 source to demonstrate the available requests and responses for sources. Read [Source Configuration](#) for information about the available parameters in the `config` object for each supported source type.

Source Object

```
{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
  "tag": "T0/Zr1FOY3A=",
  "type": "S3",
  "name": "AWS-S3_testgroup",
  "createdAt": "2023-02-17T14:32:20.640Z",
  "metadataPolicy": {
    "authTTLs": 86400000,
    "namesRefreshMs": 3600000,
    "datasetRefreshAfterMs": 3600000,
    "datasetExpireAfterMs": 10800000,
    "datasetUpdateMode": "PREFETCH_QUERIED",
    "deleteUnavailableDatasets": true,
    "autoPromoteDatasets": false
  },
}
```

```

"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"accelerationActivePolicyType": "NEVER",
"accelerationRefreshSchedule": "",
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ]
},
"roles": [
  {
    "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
    "permissions": [
      "ALTER",
      "CREATE_TABLE",
      "DROP",

```

```

        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
    ]
}
],
},
"permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
],
"checkTableAuthorizer": true,
"owner": {
    "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
    "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": false
}

```

Source Attributes

entityType String

Type of the catalog object. For sources, the entityType is `source`.

Example: source

config Object

Configuration settings for the source. The available parameters in the config object are

different for different source types. For more information, read [Source Configuration](#).

Example:

```
{ "accessKey": "EXAMPLE78HT89VS4YJEL", "accessSecret": "$DREMIO\_EXISTING\_VALUE$", "secure": true, "rootPath": "/", "enableAsync": true, "compatibilityMode": false, "isCachingEnabled": true, "maxCacheSpacePct": 100, "requesterPays": false, "enableFileStatusCheck": true, "defaultCtasFormat": "ICEBERG", "isPartitionInferenceEnabled": false, "credentialType": "ACCESS\_KEY" }
```

id String (UUID)

Unique identifier of the source.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

tag String

Unique identifier of the version of the source. Dremio changes the tag whenever the source changes and uses the tag to ensure that PUT requests apply to the most recent version of the source.

Example: T0/Zr1FOY3A=

type String

Type of source.

Valid values: `_ADX\`, `_AMAZONELASTIC\`, `_AWSGLUE\`, `_AZURE_STORAGE\`, `_BIGQUERY\`, `_DB2\`, `_DREMIOTODREMIO\`, `_DREMIO_CATALOG_EXTERNAL_V1\`, `_ELASTIC\`, `_GCS\`, `_HDFS\`, `_HIVE3\`, `_HIVE\`, `_MONGO\`, `_MSSQL\`, `_MYSQL\`, `_NAS\`, `_NESSIE\`, `_ORACLE\`, `_POSTGRES\`, `_REDSHIFT\`, `_RESTCATALOG\`, `_S3\`, `_SAPHANA\`, `_SNOWFLAKEOPENCATALOG\`, `_SNOWFLAKE\`, `_SYNAPSE\`, `_TERADATA\`, `_UNITY\`, `_VERTICA\`.

The Iceberg `_RESTCATALOG\` source type provides support for several catalogs including Apache Polaris OSS, Nessie with Iceberg REST, AWS Glue Iceberg REST, S3 Tables, Confluent Tableflow, and [Microsoft OneLake](#).

Example: S3

name String

Name of the source.

Example: AWS-S3_testgroup

createdAt String

Date and time that the source was created, in UTC format.

Example: 2023-02-17T14:32:20.640Z

[metadataPolicy](#) Object

Information about the metadata policy for the source.

Example:

```
{ "authTTLs": 86400000, "namesRefreshMs": 3600000, "datasetRefreshAfterMs": 3600000, "datasetExpireAfterMs": 10800000, "datasetUpdateMode": "PREFETCH\_QUERIED", "deleteUnavailableDatasets":
```

true,"autoPromoteDatasets": false}

accelerationGracePeriodMs Integer

Maximum age allowed for Reflection data used to accelerate queries on datasets in the source, in milliseconds. Default is `0`. For more information, read [Setting the Expiration Policy for Reflections](#).

Example: 10800000

accelerationRefreshPeriodMs Integer

Refresh period for the data in all Reflections on datasets in the source, in milliseconds. Default is `0`.

Example: 3600000

accelerationNeverExpire Boolean

Option to set an expiration for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from expiring and to override the `accelerationGracePeriodMs` setting.

Example: false

accelerationNeverRefresh Boolean

Option to set a refresh for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from refreshing and to override the `accelerationRefreshPeriodMs` setting.

accelerationActivePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `accelerationNeverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: The Reflections are refreshed at the end of every period that is defined by accelerationRefreshPeriodMs.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by accelerationRefreshSchedule.

accelerationRefreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

| Special Character | Description |

| --- | --- |

| * | Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week. |

| ? | Equivalent to *.

| , | Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.

| - | Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

children Array of Object

Information about each catalog object in the source.

Example: [{"id": "dremio:/AWS-S3_testgroup/archive.dremio.com", "path": ["AWS-S3_testgroup", "archive.dremio.com"], "type": "CONTAINER", "containerType": "FOLDER"}, {"id": "dremio:/AWS-S3_testgroup/logs_east-1", "path": ["AWS-S3_testgroup", "logs_east-1"], "type": "CONTAINER", "containerType": "FOLDER"}, {"id": "dremio:/AWS-S3_testgroup/logs_west-1", "path": ["AWS-S3_testgroup", "logs_west-1"], "type": "CONTAINER", "containerType": "FOLDER"}]

allowCrossSourceSelection Boolean

If the source is available for queries that can select from multiple sources, set to `true`. Otherwise, set to `false` (default).

Example: false

disableMetadataValidityCheck Boolean

To disable the check for expired metadata and require users to refresh manually, set to `true`. Otherwise, set to `false` (default).

note

The disableMetadataValidityCheck attribute is not supported by default. Contact Dremio Support to enable it.

Example: false

accessControlList Object

Enterprise only. Information about users and roles with access to the source and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if source-specific access control privileges are not set.

Example: {"users": [{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65", "permissions": ["VIEW_REFLECTION", "SELECT"]}], "roles": [{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions":

```
["ALTER","CREATE_TABLE","DROP","INSERT","DELETE","UPDATE","TRUNCATE","VIEW_REFLECTION","ALTER_REFLECTION","MODIFY","MANAGE_GRANTS","SELECT"]]]}
```

permissions Array of String

Enterprise-only. List of the privileges that you have on the source. Empty unless the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

```
["READ","WRITE","ALTER_REFLECTION","SELECT","ALTER","VIEW_REFLECTION","MODIFY","MANAGE_GRANTS","CREATE_TABLE","DROP","EXTERNAL_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE_SOURCE","ALL"]
```

checkTableAuthorizer Boolean

Not used. Has the value `true`.

Example: true

owner Object

Information about the source's owner.

Example:

```
{ "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365", "ownerType": "USER" }
```

accelerationRefreshOnDataChanges Boolean

If Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update, `true`. Otherwise, `false`.

Attributes of the `config` Object

The `config` object attributes vary for different source types. Read [Source Configuration](#) for information about the available parameters in the `config` object for each supported source type.

Attributes of the `metadataPolicy` Object

authTTLs Integer

Length of time to cache the privileges that the user has on the source, in milliseconds. For example, if authTTLs is set to `28800000` (8 hours), Dremio checks the user's permission status once every 8 hours. Default is `86400000` (24 hours). Minimum is `60000` (1 minute).

Example: 86400000

namesRefreshMs Integer

How often the source is refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetRefreshAfterMs Integer

How often the metadata in the source's datasets is refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

datasetExpireAfterMs Integer

Maximum age allowed for the metadata in the source's datasets, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 10800000

datasetUpdateMode String

Approach Dremio uses for updating the metadata when updating datasets in the source.

`PREFETCH`: (deprecated) Dremio updates details for all datasets in a source.

`PREFETCH_QUERIED`: Dremio updates details for previously queried objects in a source.

Example: PREFETCH_QUERIED

deleteUnavailableDatasets Boolean

If Dremio removes dataset definitions from the source when the underlying data is unavailable, set to `true` (default). Otherwise, set to `false`.

Example: true

autoPromoteDatasets Boolean

If Dremio automatically formats files into tables when a user issues a query, set to `true`. Otherwise, set to `false` (default). Available only for datalake sources, such as Amazon S3 and Hive.

Example: false

Attributes of Objects in the `children` Array

id String

Unique identifier of the child catalog object. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: dremio:/AWS-S3_testgroup/archive.dremio.com

path Array of String

Path to the child catalog object within the source, expressed as an array. The path consists of the source, followed by the name of the folder, file, or dataset itself as the last item in the array.

Example: ["AWS-S3_testgroup","archive.dremio.com"]

type String

Type of the catalog object.

Enum: CONTAINER, FILE, DATASET

Example: CONTAINER

containerType String

For catalog objects with the type `CONTAINER`, the containerType is `FOLDER`.

Example: FOLDER

datasetType String

For catalog objects with the type `DATASET`, the type of dataset. If the dataset is from an external source such as PostgreSQL, the datasetType is `DIRECT`. For tables, the datasetType is `PROMOTED`. For views, the datasetType is `VIRTUAL`.

Enum: DIRECT, PROMOTED, VIRTUAL

Example: VIRTUAL

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the source and the specific privileges each user has.

Example: [{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65","permissions": ["VIEW_REFLECTION","SELECT"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the source and the specific privileges each role has.

Example: [{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56","permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the source's owner.

Example: 4fb93af3-acc2-4b10-ad4b-64dd7070d365

ownerType String

Type of owner of the source.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise only. Unique identifier of the user or role with access to the source.

Example: ebe519ab-20e3-43ff-9b4c-b3ec590c7e65

permissions Array of String

Enterprise only. List of privileges the user or role has on the source. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["VIEW_REFLECTION","SELECT"]

Create a Source

Create a new source.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object to create. For sources, the entityType is `source`.

Example: source

config Body Object

Configuration settings for the source. The available parameters in the config object are different for different source types. For more information, read [Source Configuration](#).

Example:

```
{ "accessKey": "EXAMPLE78HT89VS4YJEL", "accessSecret": "EXAMPLEe3bcpKnAwgJ2WBpX8d9kEdhMz24guiR7L", "secure": true, "rootPath": "/", "enableAsync": true, "compatibilityMode": false, "isCachingEnabled": true, "maxCacheSpacePct": 100, "requesterPays": false, "enableFileStatusCheck": true, "defaultCtasFormat": "ICEBERG", "isPartitionInferenceEnabled": false, "credentialType": "ACCESS\_KEY" }
```

type Body String

Type of source to create.

Valid values: ``ADX``, ``AMAZONELASTIC``, ``AWSGLUE``, ``AZURE_STORAGE``, ``BIGQUERY``, ``DB2``, ``DREMIOTODREMIO``, ``DREMIO_CATALOG_EXTERNAL_V1``, ``ELASTIC``, ``GCS``, ``HDFS``, ``HIVE3``, ``HIVE``, ``MONGO``, ``MSSQL``, ``MYSQL``, ``NAS``, ``NESSIE``, ``ORACLE``, ``POSTGRES``, ``REDSHIFT``, ``RESTCATALOG``, ``S3``, ``SAPHANA``, ``SNOWFLAKEOPENCATALOG``, ``SNOWFLAKE``, ``SYNAPSE``, ``TERADATA``, ``UNITY``, ``VERTICA``.

The Iceberg ``RESTCATALOG`` source type provides support for several catalogs including Apache Polaris OSS, Nessie with Iceberg REST, AWS Glue Iceberg REST, S3 Tables, Confluent Tableflow, and [Microsoft OneLake](#).

Example: S3

name Body String

Name for the source. The name cannot include the following special characters: ``/``, ``:``, ``[``, or ``\``.

Example: AWS-S3_testgroup

metadataPolicy Body Object Optional

Information about the metadata policy for the source.

Example:

```
{ "authTTLms": 86400000, "namesRefreshMs": 3600000, "datasetRefreshAfterMs": 3600000, "datasetExpireAfterMs": 10800000, "datasetUpdateMode": "PREFETCH_QUERIED", "deleteUnavailableDatasets": true, "autoPromoteDatasets": false }
```

accelerationGracePeriodMs Body Integer Optional

Maximum age to allow for Reflection data used to accelerate queries on datasets in the source, in milliseconds. Default is ``0``. For more information, read [Setting the Expiration Policy for Reflections](#).

Example: 10800000

accelerationRefreshPeriodMs Body Integer Optional

Refresh period to use for the data in all Reflections on datasets in the source, in milliseconds. Optional if you set `accelerationActivePolicyType` to ``PERIOD``. The default setting is ``3600000`` milliseconds or one hour.

Example: 3600000

accelerationNeverExpire Body Boolean Optional

Option to set an expiration for Reflections. Default setting is ``false``. Set to ``true`` to prevent Reflections from expiring and to override the ``accelerationGracePeriodMs`` setting.

Example: false

accelerationNeverRefresh Body Boolean Optional

Option to set a refresh for Reflections. Default setting is ``false``. Set to ``true`` to prevent

Reflections from refreshing and to override the `accelerationRefreshPeriodMs` setting.

accelerationActivePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `accelerationNeverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by accelerationRefreshPeriodMs.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by accelerationRefreshSchedule.

accelerationRefreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed. Optional if you set accelerationActivePolicyType to `SCHEDULE`. The default accelerationRefreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0 N/A	
Minute	0-59 N/A	
Hour	0-23 N/A	
Day of month	N/A * ?	
Month	N/A * ?	
Days of week	1-7 or SUN-SAT , - * ?	

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
?	Equivalent to *.
,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

allowCrossSourceSelection Body Boolean Optional

If the source should be available for queries that can select from multiple sources, set to `true`. Otherwise, set to `false` (default).

Example: false

disableMetadataValidityCheck Body Boolean Optional

To disable the check for expired metadata and require users to refresh manually, set to ``true``. Otherwise, set to ``false`` (default).

note

The `disableMetadataValidityCheck` parameter is not supported by default. Contact Dremio Support to enable it.

Example: `false`

`accelerationRefreshOnDataChanges` Body Boolean

To refresh Reflections on underlying tables that are in Iceberg format in the source when new snapshots are created after an update, ``true``. Otherwise, ``false``. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. For this option to take effect, the source must support Iceberg table format, the `accelerationNeverRefresh` parameter must be set to ``false``, and the `accelerationActivePolicyType` parameter must be set to either ``PERIOD`` or ``SCHEDULE``.

`accessControlList` Body Object Optional

Enterprise only. Information about users and roles that should have access to the source and the specific privileges each user or role should have. May include an array of users, an array of roles, or both, depending on the configured access and privileges.

Example: `{ "users": [{ "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65", "permissions": ["VIEW_REFLECTION", "SELECT"] }, { "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"] }], "roles": [{ "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"] }] }`

Parameters of the ``config`` Object

The ``config`` object's parameters vary for different source types. Read [Source Configuration](#) for information about the available parameters in the ``config`` object for each supported source type.

Parameters of the ``metadataPolicy`` Object

`authTTLMs` Body Integer

Length of time to cache the privileges that the user has on the source, in milliseconds. For example, if `authTTLMs` is set to ``28800000`` (8 hours), Dremio checks the user's permission status once every 8 hours. Default is ``86400000`` (24 hours). Minimum is ``60000`` (1 minute).

Example: `86400000`

`namesRefreshMs` Body Integer Optional

How often the source should be refreshed, in milliseconds. Default is ``3600000`` (1 hour). Minimum is ``60000`` (1 minute).

Example: 3600000

`datasetRefreshAfterMs` Body Integer Optional

How often the metadata in the source's datasets should be refreshed, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 3600000

`datasetExpireAfterMs` Body Integer Optional

Maximum age to allow for the metadata in the source's datasets, in milliseconds. Default is `3600000` (1 hour). Minimum is `60000` (1 minute).

Example: 10800000

`datasetUpdateMode` Body String Optional

Approach for Dremio to take for updating the metadata when updating datasets in the source.

`PREFETCH`: (deprecated) Dremio updates details for all datasets in a source.

`PREFETCH_QUERIED`: Dremio updates details for previously queried objects in a source.

Example: `PREFETCH_QUERIED`

`deleteUnavailableDatasets` Body Boolean Optional

If Dremio should remove dataset definitions from the source when the underlying data is unavailable, set to `true` (default). Otherwise, set to `false`.

Example: true

`autoPromoteDatasets` Body Boolean Optional

If Dremio should automatically format files into tables using default options when users issue queries, set to `true`. Otherwise, set to `false` (default). Available only for datalake sources, such as Amazon S3 and Hive.

Example: false

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the source and the specific privileges each user should have.

Example: `[{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65", "permissions": ["VIEW_REFLECTION","SELECT"]}]`

roles Body Array of Object Optional

List of roles whose members should have access to the source and the specific privileges each role should have.

Example: [{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions": ["ALTER","CREATE_TABLE","DROP","INSERT","DELETE","UPDATE","TRUNCATE","VIEW_REFLECTION","ALTER_REFLECTION","MODIFY","MANAGE_GRANTS","SELECT"]}]]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Unique identifier of the user or role who should have access to the source.

Example: ebe519ab-20e3-43ff-9b4c-b3ec590c7e65

permissions Body Array of String Optional

List of privileges the user or role should have on the source. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["VIEW_REFLECTION","SELECT"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "EXAMPLEEe3bcpKnAwgJ2WBpX8d9kEdhMz24guiR7L",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "type": "S3",
  "name": "AWS-S3_testgroup",
}
```

```

"metadataPolicy": {
  "authTTLMs": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
}
}'

```

Response

```

{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",

```

```

"accessSecret": "$DREMIO_EXISTING_VALUE$",
"secure": true,
"rootPath": "/",
"enableAsync": true,
"compatibilityMode": false,
"isCachingEnabled": true,
"maxCacheSpacePct": 100,
"whitelistedBuckets": [
  "archive.dremio.com",
  "logs_east-1",
  "logs_west-1"
],
"requesterPays": false,
"enableFileStatusCheck": true,
"defaultCtasFormat": "ICEBERG",
"isPartitionInferenceEnabled": false,
"credentialType": "ACCESS_KEY"
},
"state": {
  "status": "good",
  "suggestedUserAction": "",
  "messages": []
},
"id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
"tag": "T0/Zr1FOY3A=",
"type": "S3",
"name": "AWS-S3_testgroup",
"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLms": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",

```

```

    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"checkTableAuthorizer": true,
"owner": {
  "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
  "ownerType": "USER"
}

```

```
    },  
    "accelerationRefreshOnDataChanges": false  
  }  
}
```

When you use the Catalog API to create a new source, the response includes a `state` object that describes the status of the source as shown in the example response above. The `state` object contains the following attributes:

status String

Status of the created source.

Enum: good, bad, warn

Example: good

suggestedUserAction String

Recommended action to take, if any, based on the status of the created source.

messages Array of String

Status message, if any, for the created source.

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve a Source by ID

Retrieve a source and information about its contents by specifying the source's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the source that you want to retrieve.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the source has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "source",
```

```

"config": {
  "accessKey": "EXAMPLE78HT89VS4YJEL",
  "accessSecret": "$DREMIO_EXISTING_VALUE$",
  "secure": true,
  "rootPath": "/",
  "enableAsync": true,
  "compatibilityMode": false,
  "isCachingEnabled": true,
  "maxCacheSpacePct": 100,
  "whitelistedBuckets": [
    "archive.dremio.com",
    "logs_east-1",
    "logs_west-1"
  ],
  "requesterPays": false,
  "enableFileStatusCheck": true,
  "defaultCtasFormat": "ICEBERG",
  "isPartitionInferenceEnabled": false,
  "credentialType": "ACCESS_KEY"
},
"id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
"tag": "T0/Zr1F0Y3A=",
"type": "S3",
"name": "AWS-S3_testgroup",
"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLMs": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ]
  }
]

```



```

    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ]
},
"roles": [
  {
    "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
    "permissions": [
      "ALTER",
      "CREATE_TABLE",
      "DROP",
      "INSERT",
      "DELETE",
      "UPDATE",
      "TRUNCATE",
      "VIEW_REFLECTION",
      "ALTER_REFLECTION",
      "MODIFY",
      "MANAGE_GRANTS",
      "SELECT"
    ]
  }
]
},
"permissions": [],
"checkTableAuthorizer": true,
"owner": {
  "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
  "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Source by Path

Retrieve a source and information about its contents by specifying the source's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Name of the source that you want to retrieve. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: AWS-S3_testgroup

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the source has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdkyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/AWS-S3_testgroup' \
--header 'Authorization: Bearer <<dremioAccessToken>>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
  "tag": "T0/Zr1F0Y3A=",
  "type": "S3",
  "name": "AWS-S3_testgroup",
```

```

"createdAt": "2023-02-17T14:32:20.640Z",
"metadataPolicy": {
  "authTTLMs": 86400000,
  "namesRefreshMs": 3600000,
  "datasetRefreshAfterMs": 3600000,
  "datasetExpireAfterMs": 10800000,
  "datasetUpdateMode": "PREFETCH_QUERIED",
  "deleteUnavailableDatasets": true,
  "autoPromoteDatasets": false
},
"accelerationGracePeriodMs": 10800000,
"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"children": [
  {
    "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
    "path": [
      "AWS-S3_testgroup",
      "archive.dremio.com"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_east-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_east-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "dremio:/AWS-S3_testgroup/logs_west-1",
    "path": [
      "AWS-S3_testgroup",
      "logs_west-1"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ]
}

```

```

    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"checkTableAuthorizer": true,
"owner": {
  "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
  "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": false
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Update a Source

Update the specified source.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the source to update.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

entityType Body String

Type of the catalog object to update. For sources, the entityType is `source`.

Example: source

config Body Object

Configuration settings for the source. The available parameters in the config object are different for different source types. For more information, read [Source Configuration](#).

Example:

```
{ "accessKey": "EXAMPLE78HT89VS4YJEL", "accessSecret": "EXAMPLEe3bcpKnAwgJ2WBpX8d9kEdhMz24guiR7L", "secure": true, "rootPath": "/", "enableAsync": true, "compatibilityMode": false, "isCachingEnabled": true, "maxCacheSpacePct": 100, "requesterPays": false, "enableFileStatusCheck": true, "defaultCtasFormat": "ICEBERG", "isPartitionInferenceEnabled": false, "credentialType": "ACCESS_KEY" }
```

id Body String

Unique identifier of the source to update.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

tag Body String

Unique identifier of the version of the source that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the source.

Example: T0/Zr1FOY3A=

type Body String

Type of the source.

Valid values: `ADX`, `AMAZONELASTIC`, `AWSGLUE`, `AZURE_STORAGE`, `BIGQUERY`, `DB2`, `DREMIOTODREMIO`, `DREMIO_CATALOG_EXTERNAL_V1`, `ELASTIC`, `GCS`, `HDFS`, `HIVE3`, `HIVE`, `MONGO`, `MSSQL`, `MYSQL`, `NAS`, `NESSIE`, `ORACLE`, `POSTGRES`, `REDSHIFT`, `RESTCATALOG`, `S3`, `SAPHANA`, `SNOWFLAKEOPENCATALOG`, `SNOWFLAKE`, `SYNAPSE`, `TERADATA`, `UNITY`, `VERTICA`.

The Iceberg `RESTCATALOG` source type provides support for several catalogs including Apache Polaris OSS, Nessie with Iceberg REST, AWS Glue Iceberg REST, S3 Tables, Confluent Tableflow, and [Microsoft OneLake](#).

Example: S3

name Body String

Name of the source that you want to update.

Example: AWS-S3_testgroup

metadataPolicy Body Object Optional

Information about the metadata policy for the source.

Example: {"authTTLs": 86400000,"namesRefreshMs": 3600000,"datasetRefreshAfterMs": 3600000,"datasetExpireAfterMs": 10800000,"datasetUpdateMode": "PREFETCH_QUERIED","deleteUnavailableDatasets": true,"autoPromoteDatasets": false}

accelerationGracePeriodMs Body Integer Optional

Maximum age to allow for Reflection data used to accelerate queries on datasets in the source, in milliseconds. For more information, read [Setting the Expiration Policy for Reflections](#).

Example: 10800000

accelerationRefreshPeriodMs Body Integer Optional

Refresh period to use for the data in all Reflections on datasets in the source, in milliseconds. Default is `0`.

Example: 3600000

accelerationNeverExpire Body Boolean Optional

Option to set an expiration for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from expiring and to override the `accelerationGracePeriodMs` setting.

Example: false

accelerationNeverRefresh Body Boolean Optional

Option to set a refresh for Reflections. Default setting is `false`. Set to `true` to prevent Reflections from refreshing and to override the `accelerationRefreshPeriodMs` setting.

accelerationActivePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `accelerationNeverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: The Reflections are refreshed at the end of every period that is defined by `accelerationRefreshPeriodMs`.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by `accelerationRefreshSchedule`.

accelerationRefreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the

Reflections that are defined on the source are refreshed. Optional if you set `accelerationActivePolicyType` to ``SCHEDULE``. The default `accelerationRefreshSchedule` setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
-------	----------------	----------------------------

---	---	---
-----	-----	-----

Second	0	N/A
--------	---	-----

Minute	0-59	N/A
--------	------	-----

Hour	0-23	N/A
------	------	-----

Day of month	N/A	* ?
--------------	-----	------

Month	N/A	* ?
-------	-----	------

Days of week	1-7 or SUN-SAT	, - * ?
--------------	----------------	----------

Special Character	Description
-------------------	-------------

---	---
-----	-----

*	Used to specify all values for a field. For <code>`Day of month`</code> , specifies every day of the month. For <code>`Month`</code> , specifies every month. For <code>`Days of week`</code> , specifies every day of the week.
----	--

?	Equivalent to *.
---	-------------------

,	Used to specify two or more days in the <code>`Days of week`</code> field. For example, <code>`MON,WED,FRI`</code> .
---	--

-	Used to specify ranges in the <code>`Days of week`</code> field. For example, <code>`1-3`</code> is equivalent to <code>`Sunday, Monday, and Tuesday`</code> .
---	--

Examples:

``0 0 0 * * ?`` : Refreshes every day at midnight.

``0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

``0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

`allowCrossSourceSelection` Body Boolean Optional

If the source should be available for queries that can select from multiple sources, set to ``true``. Otherwise, set to ``false`` (default).

Example: false

`disableMetadataValidityCheck` Body Boolean Optional

To disable the check for expired metadata and require users to refresh manually, set to ``true``. Otherwise, set to ``false`` (default).

note

The `disableMetadataValidityCheck` parameter is not supported by default. Contact Dremio Support to enable it.

Example: false

`accelerationRefreshOnDataChanges` Body Boolean

To refresh Reflections on underlying tables that are in Iceberg format in the source when new snapshots are created after an update, ``true``. Otherwise, ``false``. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. For this option

to take effect, the source must support Iceberg table format, the `accelerationNeverRefresh` parameter must be set to ``false``, and the `accelerationActivePolicyType` parameter must be set to either ``PERIOD`` or ``SCHEDULE``.

`accessControlList` Body String Optional

Enterprise only. Information about users and roles that should have access to the source and the specific privileges each user or role should have. May include an array of users, an array of roles, or both, depending on the configured access and privileges. To keep existing `accessControlList` settings while making other updates, duplicate the existing `accessControlList` object in the PUT request.

Example:

```
{
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": ["VIEW_REFLECTION", "SELECT"]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"]
    }
  ]
}
```

Parameters of the ``config`` Object

The ``config`` object's parameters vary for different source types. Read [Source Configuration](#) for information about the available parameters in the ``config`` object for each supported source type.

Parameters of the ``metadataPolicy`` Object

`authTTLs` Body Integer Optional

Length of time to cache the privileges that the user has on the source, in milliseconds. For example, if `authTTLs` is set to ``28800000`` (8 hours), Dremio checks the user's privilege status once every 8 hours. Default is ``86400000`` (24 hours). Minimum is ``60000`` (1 minute).

Example: 86400000

`namesRefreshMs` Body Integer Optional

How often to refresh the source, in milliseconds. Default is ``3600000`` (1 hour). Minimum is ``60000`` (1 minute).

Example: 3600000

`datasetRefreshAfterMs` Body Integer Optional

How often to refresh the metadata in the source's datasets, in milliseconds. Default is ``3600000`` (1 hour). Minimum is ``60000`` (1 minute).

Example: 3600000

`datasetExpireAfterMs` Body Integer Optional

Maximum age to allow for the metadata in the source's datasets, in milliseconds. Default is ``3600000`` (1 hour). Minimum is ``60000`` (1 minute).

Example: 10800000

datasetUpdateMode Body String Optional

Approach for Dremio to take for updating the metadata when updating datasets in the source.

``PREFETCH``: (deprecated) Dremio updates details for all datasets in a source.

``PREFETCH_QUERIED``: Dremio updates details for previously queried objects in a source.

Example: `PREFETCH_QUERIED`

deleteUnavailableDatasets Body Boolean Optional

If Dremio should remove dataset definitions from the source when the underlying data is unavailable, set to ``true`` (default). Otherwise, set to ``false``.

Example: true

autoPromoteDatasets Body Boolean Optional

If Dremio should automatically format files into tables when a user issues a query, set to ``true``. Otherwise, set to ``false`` (default). Available only for datalake sources, such as Amazon S3 and Hive.

Example: false

Parameters of the ``accessControlList`` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the source and the specific privileges each user should have.

Example: `[{"id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65", "permissions": ["VIEW_REFLECTION", "SELECT"]}]`

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the source and the specific privileges each role should have.

Example: `[{"id": "c45ff4d8-e910-4f85-89db-9b8c29188a56", "permissions": ["ALTER", "CREATE_TABLE", "DROP", "INSERT", "DELETE", "UPDATE", "TRUNCATE", "VIEW_REFLECTION", "ALTER_REFLECTION", "MODIFY", "MANAGE_GRANTS", "SELECT"]}]`

Parameters of Objects in the ``users`` and ``roles`` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the source.

Example: ebe519ab-20e3-43ff-9b4c-b3ec590c7e65

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the source. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["VIEW_REFLECTION","SELECT"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,
    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
  "tag": "T0/Zr1FOY3A=",
  "type": "S3",
  "name": "AWS-S3_testgroup",
  "metadataPolicy": {
    "authTTLMs": 86400000,
    "namesRefreshMs": 3600000,
    "datasetRefreshAfterMs": 3600000,
    "datasetExpireAfterMs": 10800000,
    "datasetUpdateMode": "PREFETCH_QUERIED",
    "deleteUnavailableDatasets": true,
    "autoPromoteDatasets": true
  },
  "accelerationGracePeriodMs": 10800000,
```

```

"accelerationRefreshPeriodMs": 3600000,
"accelerationActivePolicyType": "PERIOD",
"accelerationNeverExpire": false,
"accelerationNeverRefresh": false,
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accelerationRefreshOnDataChanges": true,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "ALTER",
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
}
}
}'

```

Response

```

{
  "entityType": "source",
  "config": {
    "accessKey": "EXAMPLE78HT89VS4YJEL",
    "accessSecret": "$DREMIO_EXISTING_VALUE$",
    "secure": true,
    "rootPath": "/",
    "enableAsync": true,
    "compatibilityMode": false,
    "isCachingEnabled": true,
    "maxCacheSpacePct": 100,

```

```

    "whitelistedBuckets": [
      "archive.dremio.com",
      "logs_east-1",
      "logs_west-1"
    ],
    "requesterPays": false,
    "enableFileStatusCheck": true,
    "defaultCtasFormat": "ICEBERG",
    "isPartitionInferenceEnabled": false,
    "credentialType": "ACCESS_KEY"
  },
  "id": "2b1be882-7012-4a99-8d6c-82e32e4562e4",
  "tag": "RfVMBBMWRvU=",
  "type": "S3",
  "name": "AWS-S3_testgroup",
  "createdAt": "2023-02-17T14:32:20.640Z",
  "metadataPolicy": {
    "authTTLms": 86400000,
    "namesRefreshMs": 3600000,
    "datasetRefreshAfterMs": 3600000,
    "datasetExpireAfterMs": 10800000,
    "datasetUpdateMode": "PREFETCH_QUERIED",
    "deleteUnavailableDatasets": true,
    "autoPromoteDatasets": true
  },
  "accelerationGracePeriodMs": 10800000,
  "accelerationRefreshPeriodMs": 3600000,
  "accelerationActivePolicyType": "PERIOD",
  "accelerationNeverExpire": false,
  "accelerationNeverRefresh": false,
  "children": [
    {
      "id": "dremio:/AWS-S3_testgroup/archive.dremio.com",
      "path": [
        "AWS-S3_testgroup",
        "archive.dremio.com"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "dremio:/AWS-S3_testgroup/logs_east-1",
      "path": [
        "AWS-S3_testgroup",
        "logs_east-1"
      ],
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "dremio:/AWS-S3_testgroup/logs_west-1",
      "path": [
        "AWS-S3_testgroup",
        "logs_west-1"
      ]
    }
  ]
}

```

```

    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  }
],
"allowCrossSourceSelection": false,
"disableMetadataValidityCheck": false,
"accessControlList": {
  "users": [
    {
      "id": "ebe519ab-20e3-43ff-9b4c-b3ec590c7e65",
      "permissions": [
        "ALTER",
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "c45ff4d8-e910-4f85-89db-9b8c29188a56",
      "permissions": [
        "ALTER",
        "CREATE_TABLE",
        "DROP",
        "INSERT",
        "DELETE",
        "UPDATE",
        "TRUNCATE",
        "VIEW_REFLECTION",
        "ALTER_REFLECTION",
        "MODIFY",
        "MANAGE_GRANTS",
        "SELECT"
      ]
    }
  ]
},
"permissions": [],
"checkTableAuthorizer": true,
"owner": {
  "ownerId": "4fb93af3-acc2-4b10-ad4b-64dd7070d365",
  "ownerType": "USER"
},
"accelerationRefreshOnDataChanges": true
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Delete a Source

Delete the specified source, including all of the source's contents.

note

If the source is in a bad state (for example, Dremio cannot authenticate to the source or the source is otherwise unavailable), only users who belong to the ADMIN role can delete the source.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the source that you want to delete.

Example: 2b1be882-7012-4a99-8d6c-82e32e4562e4

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/2b1be882-7012-4a99-8d6c-82e32e4562e4' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Catalog][(/current/reference/api/catalog/)] [Next

Source Configuration][(/current/reference/api/catalog/source/container-source-config)

Source Attributes

Create a Source

Parameters

Example

Response Status Codes

Retrieve a Source by ID

Parameters

Example

Response Status Codes

Retrieve a Source by Path

Parameters

Example

Response Status Codes

Update a Source

Parameters

Example

Response Status Codes

Delete a Source

Parameters

Example

Response Status Codes

Source:
[https://docs.dremio.com/current/reference/api/c
atalog/container-space](https://docs.dremio.com/current/reference/api/catalog/container-space)

Version: current [26.x]

On this page

Space

Use the Catalog API to retrieve information about spaces and the child objects they contain, as well as to create, update, and delete spaces.

Space Object

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
        "zips"
      ],
      "tag": "ITlp8+qyIMQ=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-08T16:24:25.084Z"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
```

```

        "MODIFY"
    ]
}
],
"roles": [
{
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
        "MODIFY"
    ]
}
]
},
"permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
],
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Space Attributes

entityType String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id String (UUID)

Unique identifier of the space.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name String

Name of the space.

Example: Example-Space

tag String

Unique identifier of the version of the space. Dremio changes the tag whenever the space changes and uses the tag to ensure that PUT requests apply to the most recent version of the space.

Example: zzOQfjY9IU0=

createdAt String

Date and time that the space was created, in UTC format.

Example: 2023-01-12T18:44:43.237Z

children Array of Object

Information about each catalog object in the space.

Example: [{"id": "8da037a1-8e50-422b-9a2b-cafb03f57c71", "path": ["Example-Space", "testfolder"], "tag": "0McuCL4MzBU=", "type": "CONTAINER", "containerType": "FOLDER"}, {"id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473", "path": ["Example-Space", "travel_testing"], "tag": "i4mnlSmHqVM=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-30T17:54:25.547Z"}, {"id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda", "path": ["Example-Space", "zips"], "tag": "ITlp8+qyIMQ=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-02-08T16:24:25.084Z"}]

accessControlList Object

Enterprise only. Information about users and roles with access to the space and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if space-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the space. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ", "WRITE", "ALTER_REFLECTION", "SELECT", "ALTER", "VIEW_REFLECTION", "MODIFY", "MANAGE_GRANTS", "CREATE_TABLE", "DROP", "EXTERNAL_QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE_SOURCE", "ALL"]

owner Object

Information about the space's owner.

Example: `{"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43","ownerType": "USER"}`

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the catalog object.

Example: 8da037a1-8e50-422b-9a2b-cafb03f57c71

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["Example-Space","testfolder"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the object.

Example: 0McuCL4MzBU=

type String

Type of the catalog object.

Enum: CONTAINER, DATASET, FILE

Example: CONTAINER

containerType String

For catalog entities with the type `CONTAINER`, the type of container.

Enum: FOLDER, FUNCTION

Example: FOLDER

datasetType String

For catalog objects in a space with the type `DATASET`, the datasetType is `VIRTUAL` (spaces cannot contain tables, only views).

Example: VIRTUAL

createdAt String

For catalog objects in a space with the type `DATASET`, date and time that the catalog object was created, in UTC format.

Example: 2023-01-30T17:54:25.547Z

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the space and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the space and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the space's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the space.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise only. Unique identifier of the user or role with access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise only. List of privileges the user or role has on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["MODIFY"]

Create a Space

Create a new space.

Method and URL

POST /api/v3/catalog

Parameters

entityType Body String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

name Body String

Name of the space. The name cannot include the following special characters: `/`, `:`, `[`, or `.`.

Example: Example-Space

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure space-specific access control privileges.

Example:

```
{ "users": [ { "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] } ], "roles": [ { "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] } ] }
```

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have.

Example:

```
[ { "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] } ]
```

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have.

Example:

```
[ { "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] } ]
```

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["MODIFY"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "space",
  "name": "Example-Space",
  "accessControllist": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
```

```

"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "MODIFY"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"permissions": [],
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve a Space by ID

Retrieve a space and information about its contents by specifying the space's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to retrieve.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
```

```

"tag": "zz0QfjY9lU0=",
"createdAt": "2023-01-12T18:44:43.237Z",
"children": [
  {
    "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
    "path": [
      "Example-Space",
      "testfolder"
    ],
    "tag": "0McuCL4MzBU=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
    "path": [
      "Example-Space",
      "travel_testing"
    ],
    "tag": "i4mn1SmHqVM=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
  },
  {
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
      "Example-Space",
      "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "MODIFY"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},

```

```
"owner": {  
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",  
  "ownerType": "USER"  
}  
}
```

Response Status Codes

200 OK

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Space by Path

Retrieve a space and information about its contents by specifying the space's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the space that you want to retrieve. The path is the name of the space.

Example: Example-Space

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the `pageToken` query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: `?maxChildren=25`

`pageToken` Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the `maxChildren` query parameter: the first request URL includes `maxChildren` set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified `maxChildren` value, the response includes a `nextPageToken` attribute. Add the `pageToken` query parameter with the `nextPageToken` value to the request URL to retrieve the next page of results. Do not remove or change the `maxChildren` query parameter when you add `pageToken` to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

`?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE`

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Example-Space' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",

```

```

    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
  },
  {
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
      "Example-Space",
      "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "MODIFY"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 204 No Content
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Update a Space

Update the specified space.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

entityType Body String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id Body String (UUID)

Unique identifier of the space to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name Body String

Name of the space to update.

Example: Example-Space

tag Body String

Unique identifier of the version of the space that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the space.

Example: zzOQfjY9IU0=

accessControlList Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. If you omit the accessControlList object in a PUT request, Dremio removes all existing user and role access settings from the space. To keep existing user and role access settings while making other updates, duplicate the existing accessControlList array in the PUT request.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]}

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have. If you omit the users object in a PUT request, Dremio removes all existing user access settings from the space. To keep existing user access settings while making other updates, duplicate the existing users array in the PUT request.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALL"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have. If you omit the roles object in a PUT request, Dremio removes all existing role access settings from the space. To keep existing role access settings while making other updates, duplicate the existing roles array in the PUT request.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["ALL"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "accessControlList": {
    "users": [
```

```
{
  "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
  "permissions": [
    "ALL"
  ]
},
"roles": [
  {
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
      "MODIFY"
    ]
  }
]
}
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "PwZ6e/axHUY=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mn1SmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
        "zips"
      ]
    }
  ]
}
```



```

    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER_REFLECTION",
        "MODIFY",
        "ALTER",
        "MANAGE_GRANTS",
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Delete a Space

Delete the specified space, including all of the space's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to delete.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Home]([current/reference/api/catalog/container-home])[Next

Folder]([current/reference/api/catalog/container-folder])

Space Attributes

Create a Space

Parameters

Example

Response Status Codes

Retrieve a Space by ID

Parameters

Example

Response Status Codes

Retrieve a Space by Path

Parameters

Example

Response Status Codes

Update a Space

Parameters

Example

Response Status Codes

Delete a Space

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/container-folder>

Version: current [26.x]

On this page

Folder

Use the Catalog API to retrieve information about folders and the child objects they contain, as well as to create, update, and delete folders.

Folder Object

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
}
```

```

"tag": "pRmJ0BQ9SFw=",
"children": [
  {
    "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
    "path": [
      "Samples",
      "samples.dremio.com",
      "zip_lookup.csv"
    ],
    "type": "FILE"
  },
  {
    "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
    "path": [
      "Samples",
      "samples.dremio.com",
      "NYC-taxi-trips-iceberg"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
    "path": [
      "Samples",
      "samples.dremio.com",
      "NYC-taxi-trips"
    ],
    "type": "DATASET",
    "datasetType": "PROMOTED"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",

```

```

    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL "
  ],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Folder Attributes

entityType String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

id String

Unique identifier of the folder. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

path Array of String

Path of the folder within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target folder itself as the last item in the array.

Example: ["Samples","samples.dremio.com"]

tag String

Unique identifier of the version of the folder. Dremio changes the tag whenever the folder changes and uses the tag to ensure that PUT requests apply to the most recent version of the folder.

Example: pRmJ0BQ9SFw=

children Array of Object

Information about each catalog object in the folder.

Example:

```
[{"id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv","path": ["Samples","samples.dremio.com","zip_lookup.csv"],"type": "FILE"}, {"id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg","path": ["Samples","samples.dremio.com","NYC-taxi-trips-iceberg"],"type": "CONTAINER","containerType": "FOLDER"}, {"id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d","path": ["Samples","samples.dremio.com","NYC-taxi-trips"],"type": "DATASET","datasetType": "PROMOTED"}]
```

accessControlList Object

Enterprise-only. Information about users and roles with access to the folder and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if folder-specific access control privileges are not set.

Example:

```
{"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}
```

permissions Array of String

Enterprise-only. List of the privileges that you have on the folder. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

```
["READ","WRITE","ALTER_REFLECTION","SELECT","ALTER","VIEW_REFLECTION","MODIFY","MANAGE_GRANTS","CREATE_TABLE","DROP","EXTERNAL_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE_SOURCE","ALL"]
```

owner Object

Information about the folder's owner.

Example:

```
{"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43","ownerType": "USER"}
```

Attributes of Objects in the `children` Array

id String

Unique identifier of the catalog object. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: `dremio:/Samples/samples.dremio.com/zip_lookup.csv`

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the catalog

object itself as the last item in the array.

Example: ["Samples","samples.dremio.com","zip_lookup.csv"]

type String

Type of the catalog object. If the object is saved within a space (including the home space), valid types are `CONTAINER` and `DATASET`. If the object is saved within a source, valid types are `CONTAINER`, `FILE`, and `DATASET`.

Example: CONTAINER

containerType String

For catalog objects with the type `CONTAINER`, the containerType is `FOLDER`.

Example: FOLDER

datasetType String

For catalog objects with the type `DATASET`, the type of dataset. For tables, the datasetType is `PROMOTED`. For views, the datasetType is `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: VIRTUAL

createdAt String

Date and time that the catalog object was created, in UTC format. The createdAt attribute is included only for `DATASET` catalog objects that are saved in folders within spaces, not within sources.

Example: 2023-01-30T17:54:25.547Z

Attributes of the `accessControlList` Object

users String

Enterprise-only. List of users with access to the folder and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}]

roles String

Enterprise-only. List of roles whose members have access to the folder and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the folder's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the folder.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise-only. Unique identifier of the user or role with access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise-only. List of privileges the user or role has on the folder. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Create a Folder

Create a new folder within a space.

note

The Catalog API cannot create new folders within sources.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

path Body Array of String

Path of the location where the folder should be created within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the target folder itself as the last item in the array. The name of the folder cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: ["Example-Space","First-Folder","New-Folder"]

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure folder-specific access control privileges.

Example: {"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["SELECT","ALTER"]}], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]}

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["SELECT","ALTER"]}]

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["SELECT"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["SELECT","ALTER"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
```

```
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  }
}'
```

Response

```
{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
```

```

      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ],
  "permissions": [],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve a Folder by ID

Retrieve a folder and information about its contents by specifying the folder's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to retrieve. If the ID is a text path, use URL encoding to replace any special characters with their UTF-8-equivalent characters, such as `%3A` for a colon; `%2F` for a forward slash; and `%20` for a space. For example, if the ID value is `dremio:/Samples/samples.dremio.com/Dremio University`, the URI-encoded ID is `dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for folders in filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/d4c2a8ba-a972-4db4-8deb-67e1ade684d1' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
```

```

"path": [
  "Samples",
  "samples.dremio.com"
],
"tag": "pRmJ0BQ9SFw=",
"children": [
  {
    "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
    "path": [
      "Samples",
      "samples.dremio.com",
      "zip_lookup.csv"
    ],
    "type": "FILE"
  },
  {
    "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
    "path": [
      "Samples",
      "samples.dremio.com",
      "NYC-taxi-trips-iceberg"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
    "path": [
      "Samples",
      "samples.dremio.com",
      "NYC-taxi-trips"
    ],
    "type": "DATASET",
    "datasetType": "PROMOTED"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
}

```

```
{,
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Folder by Path

Retrieve a folder and information about its contents by specifying the folder's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the folder that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Samples/samples.dremio.com

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query](#)

Parameters.

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
```

```

    "zip_lookup.csv"
  ],
  "type": "FILE"
},
{
  "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
  "path": [
    "Samples",
    "samples.dremio.com",
    "NYC-taxi-trips-iceberg"
  ],
  "type": "CONTAINER",
  "containerType": "FOLDER"
},
{
  "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
  "path": [
    "Samples",
    "samples.dremio.com",
    "NYC-taxi-trips"
  ],
  "type": "DATASET",
  "datasetType": "PROMOTED"
}
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Update a Folder

Update the specified folder.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

entityType Body String

Type of the catalog object. For folders, the entityType is ``folder``.

Example: folder

id Body String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

path Body Array of String

Path of the location where the folder is saved within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the target folder itself as the last item in the array.

Example: ["Example-Space", "First-Folder", "New-Folder"]

tag Body String

Unique identifier of the version of the folder that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the folder.

Example: R7COubQq8KE=

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. If you omit the `accessControlList` object in a PUT request, Dremio removes all existing user and role access settings from the folder. To keep existing user and role access settings while making other updates, duplicate the existing `accessControlList` array in the PUT request.

Example: `{"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALL"]}],"roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}`

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have. If you omit the `users` object in a PUT request, Dremio removes all existing user access settings from the folder. To keep existing user access settings while making other updates, duplicate the existing `users` array in the PUT request.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALL"]}]`

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have. If you omit the `roles` object in a PUT request, Dremio removes all existing role access settings from the folder. To keep existing role access settings while making other updates, duplicate the existing `roles` array in the PUT request.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["ALL"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALL"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  }
}'
```

Response

```
{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "vnRnYLLpCFU=",
  "children": [
    {
      "id": "d60f9258-e55a-4fc3-97b3-58c6720a70fc",
      "path": [
```

```

        "Example-Space",
        "First-Folder",
        "New-Folder",
        "NYC-trips-weather"
    ],
    "tag": "IHXU70xs80c=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "acba8595-bfcf-4126-887c-d2a19b5afb1d",
    "path": [
        "Example-Space",
        "First-Folder",
        "New-Folder",
        "short-distance-trips"
    ],
    "tag": "KYs/Qyw1ok8=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-09T19:09:58.789Z"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "ALTER_REFLECTION",
                "ALTER",
                "MANAGE_GRANTS",
                "VIEW_REFLECTION",
                "SELECT"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "SELECT"
            ]
        }
    ]
},
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Delete a Folder

Delete the specified folder, including all of the folder's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to delete. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: 598697c2-8be0-4050-9731-53563977a17d

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/598697c2-8be0-4050-9731-53563977a17d' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Space](/current/reference/api/catalog/container-space)[Next

File](/current/reference/api/catalog/file)

Folder Attributes

Create a Folder

Parameters

Example

Response Status Codes

Retrieve a Folder by ID

Parameters

Example

Response Status Codes

Retrieve a Folder by Path

Parameters

Example

Response Status Codes

Update a Folder

Parameters

Example

Response Status Codes

Delete a Folder

Parameters

Example

Response Status Codes

Source:
[https://docs.dremio.com/current/reference/api/c
atalog/container-home](https://docs.dremio.com/current/reference/api/catalog/container-home)

Version: current [26.x]

On this page

Home

Use the Catalog API to retrieve information about the home space and the child objects it contains.

Home Object

```
{
  "entityType": "home",
  "id": "87049e43-8564-4ee7-8bb6-5bdaf5bd0959",
  "name": "@user@dremio.com",
  "tag": "8S9cTZ5IsWo=",
  "children": [
    {
      "id": "1e16c0e5-c890-4f87-b1a6-ac9325aafa2c",
      "path": [
        "@user@dremio.com",
        "Business"
      ],
      "tag": "KgFBPW3+Cyc=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "a59815d0-3c21-41ad-b9bc-2ba105251fa6",
      "path": [
        "@user@dremio.com",
        "meeting_rooms_lookup"
      ],
      "tag": "0aQT64frevc=",
      "type": "DATASET",
      "datasetType": "PROMOTED",
      "createdAt": "1970-01-01T00:00:00.000Z"
    },
    {
      "id": "37401663-8666-4e00-bc03-668abb43ccd7",
      "path": [
        "@user@dremio.com",
        "NYC-trips-quarterly"
      ],
      "tag": "+H5TpLYoosY=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-07T21:26:14.385Z"
    }
  ]
}
```

Home Attributes

entityType String

Type of the catalog object. For the home space, the entityType is `home`.

Example: home

id String (UUID)

Unique identifier of the home space.

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

name String

Name of the home space. Automatically generated based on the username.

Example: @user@dremio.com

tag String

Unique identifier of the version of the home space. Dremio uses tags to ensure that PUT requests apply to the most recent version of the resource being updated. However, home spaces cannot be changed, so the tag is listed in the home space object but not used.

Example: 8S9cTZ5lsWo=

children Array of Object

Information about each catalog object in the home space.

Example: [{"id":"1e16c0e5-c890-4f87-b1a6-ac9325aafa2c","path":["@dremio","Business"],"tag":"KgFBPW3+Cyc=","type":"CONTAINER","containerType":"FOLDER"}, {"id":"a59815d0-3c21-41ad-b9bc-2ba105251fa6","path":["@dremio","meeting_rooms_lookup"],"tag":"OaQT64frevc=","type":"DATASET","datasetType":"PROMOTED","createdAt":"1970-01-01T00:00:00.000Z"}, {"id":"37401663-8666-4e00-bc03-668abb43ccd7","path":["@dremio","NYC-trips-quarterly"],"tag":"+H5TpLYoosY=","type":"DATASET","datasetType":"VIRTUAL","createdAt":"2023-02-07T21:26:14.385Z"}]

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the catalog object.

Example: 1e16c0e5-c890-4f87-b1a6-ac9325aafa2c

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the home space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["@user@dremio.com","Business"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the object.

Example: KgFBPW3+Cyc=

type String

Type of the catalog object.

Enum: CONTAINER, DATASET

Example: CONTAINER

containerType String

For catalog entities with the type `CONTAINER`, the type of container.

Enum: FOLDER, FUNCTION

Example: FOLDER

datasetType String

For catalog entities with the type `DATASET`, the type of dataset. For tables, the datasetType is `PROMOTED`. For views, the datasetType is `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: PROMOTED

createdAt String

For catalog entities with the type `DATASET`, date and time that the catalog object was created, in UTC format.

Example: 2023-02-07T21:26:14.385Z

Retrieve the Home Space by ID

Retrieve information about the home space and its contents by specifying the home space's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Body String (UUID)

Unique identifier of the home space that you want to retrieve.

Example: 87049e43-8564-4ee7-8bb6-5bdaf5bd0959

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the home space has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdkyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/63505c60-bc86-42aa-a622-24e5f22ce50b' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "home",
  "id": "87049e43-8564-4ee7-8bb6-5bdaf5bd0959",
  "name": "@user@dremio.com",
  "tag": "8S9cTZ5IsWo=",
  "children": [
    {
      "id": "1e16c0e5-c890-4f87-b1a6-ac9325aafa2c",
      "path": [
        "@user@dremio.com",
        "Business"
      ],
      "tag": "KgFBPW3+Cyc=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "a59815d0-3c21-41ad-b9bc-2ba105251fa6",
      "path": [
        "@user@dremio.com",
        "meeting_rooms_lookup"
      ],
    }
  ]
}
```

```

    "tag": "0aQT64frevc=",
    "type": "DATASET",
    "datasetType": "PROMOTED",
    "createdAt": "1970-01-01T00:00:00.000Z"
  },
  {
    "id": "37401663-8666-4e00-bc03-668abb43ccd7",
    "path": [
      "@user@dremio.com",
      "NYC-trips-quarterly"
    ],
    "tag": "+H5TpLYoosY=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-07T21:26:14.385Z"
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve the Home Space by Path

Retrieve information about the home space and its contents by specifying the home space's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the home space whose information you want to retrieve. The home space path is the username, preceded with the `@` symbol.

Example: @user@dremio.com

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the `pageToken` query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: `?maxChildren=25`

`pageToken` Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the `maxChildren` query parameter: the first request URL includes `maxChildren` set to the maximum number of child objects to include in each page of results. If the home space has more child objects than the specified `maxChildren` value, the response includes a `nextPageToken` attribute. Add the `pageToken` query parameter with the `nextPageToken` value to the request URL to retrieve the next page of results. Do not remove or change the `maxChildren` query parameter when you add `pageToken` to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

`?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE`

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/@user@dremio.com' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "home",
  "id": "87049e43-8564-4ee7-8bb6-5bdaf5bd0959",
  "name": "@user@dremio.com",
  "tag": "8S9cTZ5IsWo=",
  "children": [
    {
      "id": "1e16c0e5-c890-4f87-b1a6-ac9325aafa2c",
      "path": [
        "@user@dremio.com",
        "Business"
      ],
      "tag": "KgFBPW3+Cyc=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "a59815d0-3c21-41ad-b9bc-2ba105251fa6",
      "path": [
        "@user@dremio.com",
        "meeting_rooms_lookup"
      ],
      "tag": "0aQT64frevc=",
      "type": "DATASET",

```

```

    "datasetType": "PROMOTED",
    "createdAt": "1970-01-01T00:00:00.000Z"
  },
  {
    "id": "37401663-8666-4e00-bc03-668abb43ccd7",
    "path": [
      "@user@dremio.com",
      "NYC-trips-quarterly"
    ],
    "tag": "+H5TpLYoosY=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-07T21:26:14.385Z"
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Source](/current/reference/api/catalog/source/)[Next

Space](/current/reference/api/catalog/container-space)

Home Attributes

Retrieve the Home Space by ID

Parameters

Example

Response Status Codes

Retrieve the Home Space by Path

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/file>

Version: current [26.x]

On this page

File

Use the Catalog API to retrieve information about formatting data to a table.

File Object

```
{
  "entityType": "file",
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
  "path": [
    "Samples",
    "samples.dremio.com",
    "SF weather 2018-2019.csv"
  ]
}
```

File Attributes

entityType String

Type of the catalog object. For files, the entityType is `file`.

Example: file

id String

Unique identifier of the file. For files, the ID is the text path of the file within Dremio.

Example: dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv

path Array of String

Path of the file within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target file itself as the last item in the array.

Example: ["Samples","samples.dremio.com","SF weather 2018-2019.csv"]

Retrieve a File by Path

Retrieve information about a file by specifying its path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the file that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, replace colons with `%3A` and replace spaces with `%20`.

Example: Samples/samples.dremio.com/SF%20weather%202018-2019.csv

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/SF%20weather%2018-2019.csv' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "file",
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
  "path": [
    "Samples",
    "samples.dremio.com",
    "SF weather 2018-2019.csv"
  ]
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Was this page helpful?

[\[Previous](#)

Folder][(/current/reference/api/catalog/container-folder)][Next

Table][(/current/reference/api/catalog/table)

File Attributes

Retrieve a File by Path

Parameters

Example

Response Status Codes

Source:
[https://docs.dremio.com/current/reference/api/c
atalog/table](https://docs.dremio.com/current/reference/api/catalog/table)

Version: current [26.x]

On this page

Table

Use the Catalog API to retrieve tables, format files and folders as tables, update and refresh tables, and revert tables to files and folders.

Table Object

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "SCHEDULE",
    "refreshPeriodMs": 3600000,
    "gracePeriodMs": 10800000,
    "refreshSchedule": "0 0 8 * * ?",
    "method": "FULL",
    "neverExpire": false,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
```



```

"name": "restaurant_reviews.parquet",
"fullPath": [
  "Samples",
  "samples.dremio.com",
  "Dremio University",
  "restaurant_reviews.parquet"
],
"ctime": 0,
"isFolder": false,
"location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
"ignoreOtherFileFormats": false,
"autoCorrectCorruptDates": true
},
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",

```

```

"TRUNCATE",
"DELETE",
"UPDATE",
"EXECUTE",
"CREATE_SOURCE",
"ALL"
],
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  }
]

```

```

},
{
  "name": "stars",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "street",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "lot",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "valet",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ]
        }
      },
      {
        "name": "Accepts Credit Cards",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "Wheelchair Accessible",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  }
}

```

```

    }
  },
  {
    "name": "Price Range",
    "type": {
      "name": "BIGINT"
    }
  }
]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Table Attributes

entityType String

Type of the catalog object. For tables, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the table.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

type String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example: PHYSICAL_DATASET

path Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples","samples.dremio.com","Dremio University","restaurant_reviews.parquet"]

createdAt String

Date and time that the table was created, in UTC format.

Example: 2024-01-13T19:52:01.894Z

tag String (UUID)

Unique identifier of the version of the table. Dremio changes the tag whenever the table changes and uses the tag to ensure that PUT requests apply to the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

accelerationRefreshPolicy String

Attributes that define the acceleration refresh policy for the table.

isMetadataExpired Boolean

If true, the metadata of the table needs to be refreshed. To refresh it, run the ALTER TABLE command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the table.

If NULL, metadata has never yet been collected for the table.

lastMetadataRefreshAt String

Date and time that the table metadata was last refreshed. In UTC format. If NULL, the metadata has never yet been refreshed.

Example: 2024-01-31T09:50:01.012Z

format Object

Table format attributes.

accessControlList Object

Enterprise only. Information about users and roles with access to the table and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if table-specific access control privileges are not set.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the table. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ", "WRITE", "ALTER\ REFLECTION", "SELECT", "ALTER", "VIEW\ REFLECTION", "MODIFY", "MANAGE\ GRANTS", "CREATE\ TABLE", "DROP", "EXTERNAL\ QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE\ SOURCE", "ALL"]

owner String

Information about the table's owner.

fields Object

Attributes that represent the table schema.

approximateStatisticsAllowed Boolean

If true, `COUNT DISTINCT` queries run on the table return approximate results. Otherwise, false.

Example: {"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8","ownerType":
"USER"}

Attributes of the `accelerationRefreshPolicy` Object

activePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `neverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by refreshPeriodMs.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by refreshSchedule.

`REFRESH_ON_DATA_CHANGES`: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency.

refreshPeriodMs Integer

Refresh period for the data in all Reflections for the table, in milliseconds.

Example: 3600000

refreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

| Special Character | Description |

| --- | --- |

| * | Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week. |

| ? | Equivalent to *.

| , | Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.

| - | Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Integer

Maximum age allowed for Reflection data used to accelerate queries, in milliseconds.

Example: 10800000

method String

Approach used for refreshing the data in Reflections defined on tables that are not in the Apache Iceberg format. For more information, read [Refreshing Reflections](#).

Enum: FULL, INCREMENTAL

Example: FULL

refreshField String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Boolean

If the Reflection never expires, the value is `true`. Otherwise, the value is `false`.

Example: false

neverRefresh Boolean

If the Reflection never refreshes, the value is `true`. Otherwise, the value is `false`.

Example: false

sourceRefreshOnDataChanges Boolean

If the table's source is configured so that Reflections on tables in Iceberg format in the source will refresh when new snapshots are created after an update, `true`. Otherwise,

`false`.

Attributes of the `format` Object

type String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

Example: Parquet

name String

Table name. Dremio automatically duplicates the name of the origin file or folder to populate this value. The name of the origin file or folder cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: restaurant_reviews.parquet

fullPath Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

ctime Integer

Not used. Has the value `0`.

Example: 0

isFolder Boolean

If the value is `true`, the table was created from a folder. If the value is `false`, the table was created from a file.

Example: false

location String

Location, expressed as a string, where the table's metadata is stored within a Dremio source or space.

Example: /samples.dremio.com/Dremio University/restaurant_reviews.parquet

ignoreOtherFileFormats Boolean

If true, Dremio ignores all non-Parquet files in the related folder structure, and the promoted table works as if only Parquet files are in the folder structure. Otherwise, false. Included only for Parquet folders.

Example: false

metaStoreType String

Not used. Has the value `HDFS`.

Example: HDFS

parquetDataFormat Object

Information about data format for Parquet tables.

dataFormatTypeList Array of String

List of data format types in the table. Included only for Iceberg tables, and `PARQUET` is the only valid value.

Example: ["PARQUET"]

sheetName String

For tables created from files that contain multiple sheets, the name of the sheet used to create the table.

Example: location_1

extractHeader Boolean

For tables created from files, the value is `true` if Dremio extracted the table's column names from the first line of the file. Otherwise, the value is `false`.

Example: false

hasMergedCells Boolean

For tables created from files, the value is `true` if Dremio expanded merged cells in the file when creating the table. Otherwise, the value is `false`.

Example: true

fieldDelimiter String

Character used to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character.

quote String

Character used for quotation marks in the table. May be `"` for a double quote (default), `'` for a single quote, or a custom character.

comment String

Character used to indicate comments in the table. May be `#` for a number sign (default) or a custom character.

escape String

Character used to indicate an escape in the table. May be `"` for a double quote (default), `` for a back quote, `\\` for a backward slash, or a custom character.

lineDelimiter String

Character used to indicate separate lines in the table. May be `\r\n` for a carriage

return and a new line (default), ``\n`` for a new line, or a custom character.

`skipFirstLine` Boolean

If Dremio skipped the first line in the file or folder when creating the table, the value is ``true``. Otherwise, the value is ``false``.

Example: false

`autoGenerateColumnNames` Boolean

If Dremio used the existing column names in the file or folder for the table columns, the value is ``true``. Otherwise, the value is ``false``.

Example: true

`trimHeader` Boolean

If Dremio trimmed column names to a specific number of characters when creating the table, the value is ``true``. Otherwise, the value is ``false``.

Example: true

`autoCorrectCorruptDates` Boolean

If Dremio automatically corrects corrupted date fields in the table, the value is ``true``. Otherwise, the value is ``false``.

Example: true

Attributes of the ``accessControlList`` Object

users Array of Object

Enterprise only. List of users with access to the table and the specific privileges each user should have.

Example: `[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT", "ALTER", "MANAGE\ GRANTS"]}]`

roles Array of Object

Enterprise only. List of roles whose members have access to the table and the specific privileges each role has.

Example: `[{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT", "ALTER"]}]`

Attributes of Objects in the ``users`` and ``roles`` Arrays

`id` String

Enterprise only. Unique identifier of the user or role with access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the table's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the table.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the table field.

Example: review_count

type Object

Information about the table field.

Attributes of the `type` Object

name String

Name of the table field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: BIGINT

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 0

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema appears only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: Parking

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Attributes of the `parquetDataFormat` Object

type String

Type of data in the table. Within the parquetDataFormat object, the only valid type is `Parquet`.

Example: Parquet

ctime Integer

Not used. Has the value `0`.

Example: 0

isFolder Boolean

If the value is `true`, the table was created from a folder. If the value is `false`, the table was created from a file.

Example: true

autoCorrectCorruptDates Boolean

If the value is `true`, Dremio automatically corrects corrupted date fields in the table. Otherwise, the value is `false`.

Example: true

Format a File or Folder as a Table

Format a file or folder as a table so that you can query the data in Dremio.

note

To format a folder, all files in the folder must be the same format.

Method and URL

```
POST /api/v3/catalog/{id}
```

Parameters

`id` Path String

Unique identifier of the file or folder you want to format. The ID can be a UUID or a text path. If the ID is a text path, use URL encoding to replace special characters with their UTF-8-equivalent characters: ``%3A`` for a colon; ``%2F`` for a forward slash, and ``%20`` for a space. For example, if the ID value is ``dremio:/Samples/samples.dremio.com/Dremio University``, the URI-encoded ID is ``dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University``.

Example: `c590ed7f-7142-4e1f-ba7d-94173afdc9a3`

`entityType` Body String

Type of the catalog object. To format a file or folder as a table, the `entityType` is ``dataset``.

`path` Body Array of String

Path of the file or folder you want to format, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the file or folder itself as the last item in the array. Get the path from the file or folder's `children` object in the response to a [Folder](#) request.

Example: `["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]`

`type` Body String

Type of dataset. For tables, the type is ``PHYSICAL_DATASET``.

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

[format](#) Body String

Parameters that describe how to format the file or folder.

[accessControlList](#) Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: `{"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions":`

```
["SELECT", "ALTER", "MANAGE\ GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]]}
```

Parameters of the `accelerationRefreshPolicy` Object

activePolicyType Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the neverRefresh parameter must be set to `false`.

The possible values are:

- `NEVER`: The Reflections are never refreshed.
- `PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by refreshPeriodMs.
- `SCHEDULE`: The Reflections are refreshed according to the schedule that is set by refreshSchedule.
- `REFRESH_ON_DATA_CHANGES`: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in Iceberg format.

refreshPeriodMs Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set activePolicyType to `PERIOD`. The default setting is `3600000` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

refreshSchedule Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set activePolicyType to `SCHEDULE`. The default refreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0 N/A	
Minute	0-59 N/A	
Hour	0-23 N/A	
Day of month	N/A * ?	
Month	N/A * ?	
Days of week	1-7 or SUN-SAT , - * ?	

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.

| ? | Equivalent to *. |

| , | Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`. |

| - | Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`. |

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in AWS Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the file or folder. All files in the folder must be the same format.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

ignoreOtherFileFormats Body Boolean Optional

If Dremio should ignore all non-Parquet files in the related folder structure so that the promoted table works as if only Parquet files are in the folder structure, set to `true`. Otherwise, set to `false` (default). Optional for Parquet folders.

Example: false

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the file or folder when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Excel and Text types.

Example: true

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the file when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Excel type.

Example: true

sheetName Body String Optional

For tables created from Excel files that contain multiple sheets, the name of the sheet to use to create the table. Default is the first sheet in the file (for files that contain multiple sheets).

Example: location_1

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character. Optional for files or folders of the Text type.

quoteBody String Optional

Character to use for quotes in the table. May be `"` for a double quote (default), `'` for a single quote, or a custom character. Optional for files or folders of the Text type.

commentBody String Optional

Character to use to indicate comments in the table. May be `#` for a number sign (default) or a custom character. Optional for files or folders of the Text type.

escapeBody String Optional

Character used to indicate an escape in the table. May be `"` for a double quote (default), `` ` `` for a back quote, `\\` for a backward slash, or a custom character. Optional for files or folders of the Text type.

lineDelimiterBody String Optional

Character used to indicate separate lines in the table. May be `\r\n` for a carriage return and a new line (default), `\n` for a new line, or a custom character. Optional for files or folders of the Text type.

autoGenerateColumnNamesBody Boolean Optional

If Dremio should use the existing column names in the file or folder for the table columns, set to `true` (default). Otherwise, set to `false`. Optional for files or folders of the Text type.

Example: true

trimHeaderBody Boolean Optional

If Dremio should trim column names to a specific number of characters when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT", "ALTER"]}]]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Samples",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Parquet"
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ]
  }
}
```

```

    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
}
}'

```

Response

```

{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": "false",

```

```

    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  },
  "fields": [
    {
      "name": "_id",
      "type": {
        "name": "VARCHAR"
      }
    },
    {
      "name": "name",
      "type": {
        "name": "VARCHAR"
      }
    },
    {
      "name": "city",
      "type": {
        "name": "VARCHAR"
      }
    }
  ],
  {

```

```

    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "attributes",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "Parking",
          "type": {
            "name": "STRUCT",
            "subSchema": [
              {
                "name": "garage",
                "type": {
                  "name": "BOOLEAN"
                }
              }
            ]
          }
        },
        {
          "name": "street",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },
  {
    "name": "lot",

```

```

        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "valet",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  },
  {
    "name": "Accepts Credit Cards",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Wheelchair Accessible",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Price Range",
    "type": {
      "name": "BIGINT"
    }
  }
]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Example Request for Excel format type

```

curl -X POST \
'https://{hostname}/api/v3/catalog/dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20U\
niversity%2Foracle-departments.xlsx' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",

```

```

"path": [
  "Samples",
  "samples.dremio.com",
  "Dremio University",
  "oracle-departments.xlsx"
],
"type": "PHYSICAL_DATASET",
"format": {
  "type": "Excel",
  "extractHeader": true,
  "hasMergedCells": true,
  "sheetName": "Sheet1"
}
}'

```

Example Request for Text format type

```

curl -X POST 'https://{hostname}/api/v3/catalog/6ba3bd6e-fd27-4572-a535-77e1548283b3' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Text",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": false
  }
}'

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a Table by ID

Retrieve a table by specifying the table's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to retrieve.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
```



```

"accelerationRefreshPolicy": {
  "activePolicyType": "PERIOD",
  "refreshPeriodMs": 3600000,
  "refreshSchedule": "0 56 18 * * *",
  "gracePeriodMs": 259200000,
  "method": "FULL",
  "neverExpire": true,
  "neverRefresh": false,
  "sourceRefreshOnDataChanges": false
},
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"format": {
  "type": "Parquet",
  "name": "restaurant_reviews.parquet",
  "fullPath": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "ctime": 0,
  "isFolder": false,
  "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
  "ignoreOtherFileFormats": false,
  "autoCorrectCorruptDates": true
},
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
}

```

```

},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  }
]

```

```

},
{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "street",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "lot",
              "type": {
                "name": "BOOLEAN"
              }
            },
            {
              "name": "valet",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ]
        }
      }
    ],
    "name": "Accepts Credit Cards",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Wheelchair Accessible",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Price Range",
    "type": {
      "name": "BIGINT"
    }
  }
}

```

```

    }
  }
]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Table by Path

Retrieve a table by specifying the table's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Table's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" table in the "samples.dremio.com" folder within the source "Samples," the path is `Samples/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example:

Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
```

```

"location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
"ignoreOtherFileFormats": false,
"autoCorrectCorruptDates": true
},
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  }
]

```

```

},
{
  "name": "state",
  "type": {
    "name": "VARCHAR"
  }
},
{
  "name": "categories",
  "type": {
    "name": "LIST",
    "subSchema": [
      {
        "type": {
          "name": "VARCHAR"
        }
      }
    ]
  }
},
{
  "name": "review_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "stars",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {
              "name": "garage",
              "type": {
                "name": "BOOLEAN"
              }
            }
          ]
        }
      },
      {
        "name": "street",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  }
},

```

```

        {
            "name": "lot",
            "type": {
                "name": "BOOLEAN"
            }
        },
        {
            "name": "valet",
            "type": {
                "name": "BOOLEAN"
            }
        }
    ]
}
},
{
    "name": "Accepts Credit Cards",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Wheelchair Accessible",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Price Range",
    "type": {
        "name": "BIGINT"
    }
}
]
}
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Update a Table

Update the specified table in Dremio.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

entityType Body String

Type of the catalog object. For tables, the entityType is `dataset`.

id Body String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

path Body Array of String

Path of the table that you want to update, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array. Get the path from the table's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

tag Body String (UUID) Optional

Unique identifier of the version of the table that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the table. If you do not provide a tag, Dremio automatically updates the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

type Body String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example:

accelerationRefreshPolicy Object

Attributes that define the acceleration refresh policy for the table.

format Body String

Parameters that describe the table's format.

accessControlList Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT","ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT","ALTER","MANAGE\ GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]}

Parameters of the `accelerationRefreshPolicy` Object

activePolicyType Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the `neverRefresh` parameter must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by `refreshPeriodMs`.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

`REFRESH_ON_DATA_CHANGES`: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in Iceberg format.

refreshPeriodMs Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set `activePolicyType` to `PERIOD`. The default setting is `3600000` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

refreshSchedule Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set

activePolicyType to `SCHEDULE`. The default refreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
?	Equivalent to *.
,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in AWS Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full

refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the table, set to `true`. Otherwise, set to `false` (default). Optional for Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Excel and Text types.

Example: true

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the table, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Excel type.

Example: true

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character. Optional for tables created from files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be `"` for a double quote (default), `` for a single quote, or a custom character. Optional for tables created from files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments for the table. May be `#` for a number sign (default) or a custom character. Optional for tables created from files or folders of the Text type.

escape Body String Optional

Character to use to indicate an escape for the table. May be `\"` for a double quote (default), `` for a back quote, `\\` for a backward slash, or a custom character. Optional for tables created from files or folders of the Text type.

lineDelimiter Body String Optional

Character to use to indicate separate lines for the table. May be `\r\n` for a carriage return and a new line (default), `\n` for a new line, or a custom character. Optional for tables created from files or folders of the Text type.

Example:

autoGenerateColumnNames Body Boolean Optional

If Dremio should use the existing column names for the table columns, set to `true` (default). Otherwise, set to `false`. Optional for tables created from files or folders of the Text type.

Example: true

trimHeader Body Boolean Optional

If Dremio should trim column names to a specific number of characters when updating the table, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example:

```
[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions":
```

```
["SELECT","ALTER","MANAGE\GRANTS"]}]}
```

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example:

```
{ "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"] }
```

Parameters of Objects in the `users` and `roles` Arrays

id Body String

Enterprise only. Unique identifier of the user or role that should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/dbale4fe-6351-44d2-a3e0-7aa20e782bf3' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "dbale4fe-6351-44d2-a3e0-7aa20e782bf3",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Text",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  }
}
```

```
}'
```

Response

```
{
  "entityType": "dataset",
  "id": "dbale4fe-6351-44d2-a3e0-7aa20e782bf3",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "createdAt": "2024-01-23T21:26:59.568Z",
  "tag": "fc1707df-35a1-45c1-87d7-5f66fb11a729",
  "format": {
    "type": "Text",
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/airbnb_listings.csv",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  },
  "accessControlList": {},
  "owner": {
    "ownerId": "c590ed7f-7142-4e1f-ba7d-94173afdc9a3",
    "ownerType": "USER"
  },
  "fields": [
    {
      "name": "id",
      "type": {
        "name": "VARCHAR"
      }
    },
    {
      "name": "listing_url",
      "type": {
        "name": "VARCHAR"
      }
    },
    {
      "name": "scrape_id",
      "type": {
        "name": "VARCHAR"
      }
    }
  ]
}
```

```

    }
  },
  {
    "name": "last_scraped",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "summary",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "reviews_per_month",
    "type": {
      "name": "VARCHAR"
    }
  }
},
"approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Refresh the Reflections on a Table

Refresh the Reflections associated with the specified table.

note

Refreshing a table's Reflections does not refresh its metadata. Read [Refreshing Metadata](#) to learn how to refresh table metadata. Read [Refreshing Reflections](#) for more

information about refreshing Reflections.

Method and URL

```
POST /api/v3/catalog/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example

Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Revert a Table to a File or Folder

Revert a table in a source to change the data in the table back to its original format, file or folder. For more information, read [Formatting Data to a Table](#) and [Removing Formatting on Data](#)

note

If a table is saved in your home space, the revert request will delete the table entirely. The revert endpoint only changes a table back to a file or folder if the table is saved in a source.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to revert to a file or folder.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[\[Previous](#)

[File\]\(/current/reference/api/catalog/file\)\[Next](#)

[User-Defined Function\]\(/current/reference/api/catalog/user-defined-function\)](#)

[Table Attributes](#)

[Format a File or Folder as a Table](#)

[Parameters](#)

[Example](#)

Response Status Codes

Retrieve a Table by ID

Parameters

Example

Response Status Codes

Retrieve a Table by Path

Parameters

Example

Response Status Codes

Update a Table

Parameters

Example

Response Status Codes

Refresh the Reflections on a Table

Parameters

Example

Response Status Codes

Revert a Table to a File or Folder

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/user-defined-function>

Version: current [26.x]

On this page

User-Defined Function

Use the Catalog API to retrieve information about user-defined functions (UDFs), as well as to create, update, and delete UDFs.

User-Defined Function Object

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\"
WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\"
DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  },
  "permissions": [],
  "owner": {
    "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
    "ownerType": "USER"
  }
}
```

User-Defined Function Attributes

entityType String

Type of the catalog object. For user-defined functions, the entityType is `function`.

id String (UUID)

Unique identifier of the user-defined function.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

path Array of String

Path of the user-defined function within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the function itself as the last item in the array.

Example: ["team_folder","test_subfolder","filter_domain_orderdates"]

tag String

Unique identifier of the version of the user-defined function. Dremio changes the tag whenever the function changes and uses the tag to ensure that PUT requests apply to the most recent version of the function.

Example: qBWpD7x6+Ws=

createdAt String

Date and time at which the user-defined function was created, in UTC format.

Example: 2024-08-01T20:20:38.547Z

lastModified String

Date and time at which the user-defined function was last modified, in UTC format.

Example: 2024-08-01T20:20:38.547Z

isScalar Boolean

If the user-defined function is a scalar function, `true`. If the user-defined function is a tabular function, `false`.

Example: false

functionArgList String

The user-defined function's arguments and their data types. If the function includes multiple arguments, the arguments are separated with a comma.

note

In response objects, the functionArgList value may contain aliases for data types, such as `CHARACTER VARYING` (an alias for `VARCHAR`).

Example: "domain" CHARACTER VARYING, "orderdate" DATE

functionBody String

The statement that the user-defined function executes.

Example: SELECT "name", "email", "order_date" FROM "customer_data" WHERE LOWER("email") LIKE '% ' || LOWER(domain) AND "order_date" >= orderdate

returnType String

The data type of the result that the function returns (for scalar functions) or of each column that the function returns, separated by commas (for tabular functions).

Example: "name" CHARACTER VARYING, "email" CHARACTER VARYING, "order_date" DATE

accessControlList Object

Enterprise only. Information about users and roles with access to the user-defined function and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if function-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER", "EXECUTE"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER", "EXECUTE"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the user-defined function. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [User-Defined Function \(UDF\) Privileges](#).

owner Object

Information about the user-defined function's owner.

Example: {"ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349", "ownerType": "USER"}

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the user-defined function and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER", "EXECUTE"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the user-defined function and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER", "EXECUTE"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the user-defined function's owner.

Example: 4740ab48-39c6-434c-9086-8f6e52e65349

ownerType String

Type of owner of the user-defined function.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the user-defined function.

Example: 4740ab48-39c6-434c-9086-8f6e52e65349

permissions Array of String

Enterprise only. List of privileges the user or role has on the user-defined function. For more information, read [User-Defined Function \(UDF\) Privileges](#).

Enum: ALTER, EXECUTE, MANAGE_GRANTS, OWNERSHIP

Example: ["ALTER","EXECUTE"]

Create a User-Defined Function

Create a new user-defined function.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For user-defined functions, the entityType is `function`.

path Body Array of String

Path where you want to create the user-defined function within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the function itself as the last item in the array.

Example: ["team_folder","test_subfolder","filter_domain_orderdates"]

isScalar Body Boolean

If the user-defined function is a scalar function, `true`. If the user-defined function is a tabular function, `false`.

Example: false

functionArgList Body String

The name of each argument in the user-defined function and the argument's data type. Separate the name and data type with a single space. If the function includes multiple arguments, separate the arguments with a comma.

note

In response objects, the **functionArgList** value may contain aliases for data types, such as ``CHARACTER VARYING`` (an alias for ``VARCHAR``).

Example: domain VARCHAR, orderdate DATE

functionBody Body String

The statement that the user-defined function should execute.

Example: `SELECT name, email, order_date FROM customer_data WHERE LOWER(email) LIKE '% ' || LOWER(domain) AND order_date >= orderdate`

returnType Body String

The data type of each column that the user-defined function should return.

Example: name VARCHAR, email VARCHAR, order_date DATE

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the user-defined function and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure function-specific access control privileges.

Example: `{"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER", "EXECUTE"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER", "EXECUTE"]}]}`

Parameters of the ``accessControlList`` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the user-defined function and the specific privileges each user should have.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER", "EXECUTE"]}]`

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the user-defined function and the specific privileges each role should have.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER", "EXECUTE"]}]`

Parameters of Objects in the ``users`` and ``roles`` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the user-defined function.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the user-defined function. For more information, read [User-Defined Function \(UDF\) Privileges](#).

Enum: ALTER, EXECUTE, MANAGE_GRANTS, OWNERSHIP

Example: ["ALTER","EXECUTE"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "function",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "isScalar": false,
  "functionArgList": "domain VARCHAR, orderdate DATE",
  "functionBody": "SELECT name, email, order_date FROM customer_data WHERE LOWER(email)
LIKE '%' || LOWER(domain) AND order_date >= orderdate",
  "returnType": "name VARCHAR, email VARCHAR, order_date DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  }
}
```

```
}'
```

Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\"
WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\"
DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  },
  "permissions": [],
  "owner": {
    "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
    "ownerType": "USER"
  }
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

409 Conflict

Retrieve a User-Defined Function by ID

Retrieve a user-defined function and information about its contents by specifying the function's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user-defined function that you want to retrieve.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/1568aa06-4eac-48cf-bc30-2aa3053c2840' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
```

```

"isScalar": false,
"functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
"functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\"
WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >= orderdate",
"returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\"
DATE",
"accessControllist": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER",
        "EXECUTE"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "ALTER",
        "EXECUTE"
      ]
    }
  ]
},
"permissions": [],
"owner": {
  "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a User-Defined Function by Path

Retrieve a user-defined function and information about its contents by specifying the function's path.

Method and URL

GET /api/v3/catalog/by-path/{path}

Parameters

path Path String

Path of the user-defined function within Dremio. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the function itself. Separate each level of the path with a forward slash.

Example: team_folder/test_subfolder/filter_domain_orderdates

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/team_folder/test_subfolder/filter_domain_orderdates' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBWpD7x6+Ws=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-01T20:20:38.547Z",
  "isScalar": false,
  "functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
  "functionBody": "SELECT \"name\", \"email\", \"order_date\" FROM \"customer_data\" WHERE LOWER(\"email\") LIKE '%\" || LOWER(domain) AND \"order_date\" >= orderdate",
  "returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING, \"order_date\" DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  }
}
```

```

    ]
  }
],
"roles": [
  {
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
      "ALTER",
      "EXECUTE"
    ]
  }
]
},
"permissions": [],
"owner": {
  "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Update a User-Defined Function

Update the specified user-defined function.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user-defined function that you want to update.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

entityType Body String

Type of the catalog object. For user-defined functions, the entityType is `function`.

id Body String (UUID)

Unique identifier of the user-defined function that you want to update.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

path Body Array of String

Path of the user-defined function within Dremio, expressed as an array. The path consists of the source or user-defined function, followed by any folder and subfolders, followed by the name of the function itself as the last item in the array.

Example: ["team_folder","test_subfolder","filter_domain_orderdates"]

tag Body String

Unique identifier of the version of the user-defined function that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the user-defined function.

Example: qBWpD7x6+Ws=

isScalar Body Boolean

If the user-defined function is a scalar function, `true`. If the user-defined function is a tabular function, `false`.

Example: false

functionArgList Body String

The name of each argument in the user-defined function and the argument's data type. Separate the name and data type with a single space. If the function includes multiple arguments, separate the arguments with a comma.

note

In response objects, the functionArgList value may contain aliases for data types, such as `CHARACTER VARYING` (an alias for `VARCHAR`).

Example: domain VARCHAR, orderdate DATE

functionBody Body String

The statement that the user-defined function should execute.

Example: SELECT name, email, phone_number, order_date FROM customer_data WHERE LOWER(email) LIKE '%' || LOWER(domain) AND order_date >= orderdate

returnType Body String

The data type of each column that the user-defined function should return.

Example: name VARCHAR, email VARCHAR, phone_number VARCHAR, order_date DATE

accessControlList Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the user-defined function and the specific privileges each user or role should have. If you omit the accessControlList object in a PUT request, Dremio removes all existing user

and role access settings from the function. To keep existing user and role access settings while making other updates, duplicate the existing `accessControlList` array in the PUT request.

Example: `{"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER","EXECUTE", "MANAGE\ _GRANTS"]}], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER","EXECUTE"]}]}`

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the user-defined function and the specific privileges each user should have. If you omit the `users` object in a PUT request, Dremio removes all existing user access settings from the function. To keep existing user access settings while making other updates, duplicate the existing `users` array in the PUT request.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["ALTER","EXECUTE", "MANAGE\ _GRANTS"]}]`

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the user-defined function and the specific privileges each role should have. If you omit the `roles` object in a PUT request, Dremio removes all existing role access settings from the function. To keep existing role access settings while making other updates, duplicate the existing `roles` array in the PUT request.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["ALTER","EXECUTE"]}]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the user-defined function.

Example: `737a038f-c6cd-4fd3-a77a-59f692727ba5`

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the user-defined function. For more information, read [User-Defined Function \(UDF\) Privileges](#).

Enum: ALTER, EXECUTE, MANAGE\ _GRANTS, OWNERSHIP

Example: `["ALTER","EXECUTE", "MANAGE\ _GRANTS"]`

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/1568aa06-4eac-48cf-bc30-2aa3053c2840' \
```



```
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "qBwPD7x6+Ws=",
  "isScalar": false,
  "functionArgList": "domain VARCHAR, orderdate DATE",
  "functionBody": "SELECT name, email, phone_number, order_date FROM customer_data WHERE
LOWER(email) LIKE '%" || LOWER(domain) AND order_date >= orderdate",
  "returnType": "name VARCHAR, email VARCHAR, phone_number VARCHAR, order_date DATE",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALTER",
          "EXECUTE",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "ALTER",
          "EXECUTE"
        ]
      }
    ]
  }
}'
```

Response

```
{
  "entityType": "function",
  "id": "1568aa06-4eac-48cf-bc30-2aa3053c2840",
  "path": [
    "team_folder",
    "test_subfolder",
    "filter_domain_orderdates"
  ],
  "tag": "4RuPbmWPoa9=",
  "createdAt": "2024-08-01T20:20:38.547Z",
  "lastModified": "2024-08-07T17:17:17.360Z",
  "isScalar": false,
```

```

"functionArgList": "\"domain\" CHARACTER VARYING, \"orderdate\" DATE",
"functionBody": "SELECT \"name\", \"email\", \"phone_number\", \"order_date\" FROM
\"customer_data\" WHERE LOWER(\"email\") LIKE '%' || LOWER(domain) AND \"order_date\" >=
orderdate",
"returnType": "\"name\" CHARACTER VARYING, \"email\" CHARACTER VARYING,
\"phone_number\" CHARACTER VARYING, \"order_date\" DATE",
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER",
        "EXECUTE",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "ALTER",
        "EXECUTE"
      ]
    }
  ]
},
"permissions": [],
"owner": {
  "ownerId": "4740ab48-39c6-434c-9086-8f6e52e65349",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Delete a User-Defined Function

Delete the specified user-defined function.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the user-defined function that you want to delete.

Example: 1568aa06-4eac-48cf-bc30-2aa3053c2840

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/1568aa06-4eac-48cf-bc30-2aa3053c2840' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Table](/current/reference/api/catalog/table)[Next

View](/current/reference/api/catalog/view)

User-Defined Function Attributes

Create a User-Defined Function

Parameters

Example

-

Response Status Codes

Retrieve a User-Defined Function by ID

Parameters

Example

Response Status Codes

Retrieve a User-Defined Function by Path

Parameters

Example

Response Status Codes

Update a User-Defined Function

Parameters

Example

Response Status Codes

Delete a User-Defined Function

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/view>

Version: current [26.x]

On this page

View

Use the Catalog API to retrieve, create, update, and delete views.

View Object

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
```

```

"createdAt": "2022-11-17T18:31:23.236Z",
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",
  "TRUNCATE",
  "DELETE",
  "UPDATE",

```

```

    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  },
  "fields": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "passenger_payment_method",
      "type": {
        "name": "STRUCT",
        "subSchema": [
          {
            "name": "cash",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "credit-debit",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "payment-app",
            "type": {
              "name": "BOOLEAN"
            }
          },
          {
            "name": "other",
            "type": {
              "name": "BOOLEAN"
            }
          }
        ]
      }
    }
  ]
},
{

```

```

    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}

```

View Attributes

entityType String

Type of the catalog object. For views, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the view.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

type String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

Example: VIRTUAL\DATASET

path Array of String

Path of the view within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the view itself as the last item in the array.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

createdAt String

Date and time that the view was created, in UTC format.

Example: 2022-11-17T18:31:23.236Z

isMetadataExpired Boolean

If true, the metadata of the tables that the view is defined on needs to be refreshed. To refresh it, run the ALTER VIEW command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the view.

If NULL, metadata has never yet been collected for the tables that the view is defined on.

lastMetadataRefreshAt String

Date and time that the metadata of the tables that the view is defined on was last refreshed. In UTC format.

Example: 2024-01-31T09:50:01.012Z

tag String (UUID)

Unique identifier of the version of the view. Dremio changes the tag whenever the view changes and uses the tag to ensure that PUT requests apply to the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql String

SQL query used to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Array of String

Context for the SQL query used to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Object

Enterprise only. Information about users and roles with access to the view and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if view-specific access control privileges are not set.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"] }, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"] }], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the view. Only appears in the

response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

```
["READ","WRITE","ALTER\_REFLECTION","SELECT","ALTER","VIEW\_REFLECTION","MODIFY","MANAGE\GRANTS","CREATE\TABLE","DROP","EXTERNAL_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE\SOURCE","ALL"]
```

owner String

Information about the view's owner.

Example: {"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8","ownerType": "USER"}

fields Array of Object

Attributes that represent the dataset schema.

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the view and the specific privileges each user has.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT","ALTER"]},{ "id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT","ALTER", "MANAGE\GRANTS"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the view and the specific privileges each role has.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the view. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the view's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the view.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the view field.

Example: pickup_datetime

type Object

Information about the view field.

Attributes of the `type` Object

name String

Name of the view field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: TIMESTAMP

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 2

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema is listed only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: cash

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Create a View

Create a view from a table in Dremio.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the view itself as the last item in the array. The name of the view cannot include the following special characters: `/`, `:`, `[`, or `]`. Views can only be created in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

sql Body String

SQL query to use to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Body Array of String

Context for the SQL query to use to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: `{ "users": [{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"] }, { "id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"] }], "roles": [{ "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"] }] }`

Parameters of the `accessControlList` Object

users [Body] Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: `[{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"] }, { "id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"] }]`

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: `[{ "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"] }]`

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: `c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3`

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: `["SELECT", "ALTER"]`

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
```

```

"path": [
  "Business",
  "Transportation",
  "NYC-taxi-trips-short-distance"
],
"type": "VIRTUAL_DATASET",
"sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
"sqlContext": [
  "Samples",
  "samples.dremio.com"
],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
}
}'

```

Response

```

{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],

```

```

"createdAt": "2022-11-17T18:31:23.236Z",
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  }
]

```

```

},
{
  "name": "passenger_payment_method",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "cash",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "credit-debit",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "payment-app",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "other",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  }
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {

```

```
    "name": "DOUBLE"
  }
}
]
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a View by ID

Retrieve a view by specifying the view's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to retrieve.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <<dremioAccessToken>>' \
--header 'Content-Type: application/json'
```


Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  },
  "fields": [
    {
```

```

    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "other",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  }

```

```

    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a View by Path

Retrieve a view by specifying the view's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

View's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" view in the "samples.dremio.com" folder within the space "Transportation," the path is `Transportation/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Business/Transportation/NYC-taxi-trips-short-distance

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Business/Transportation/NYC-taxi-trips-short-distance' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ]
  }
}
```

```

    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "other",

```

```

        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
}
]
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Update a View

Update a view in Dremio.

Method and URL

Parameters

id Path String (UUID)

Unique identifier of the view that you want to update.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the updated view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the name for the view itself as the last item in the array. Views can only be saved in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

tag Body String (UUID) Optional

Unique identifier of the version of the view that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the view. If you do not provide a tag, Dremio automatically updates the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql Body String

SQL query to use to update the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC

sqlContext Body Array of String

Context for the SQL query to use for the updated view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\\GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "type": "VIRTUAL_DATASET",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
```



```
"Samples",
"samples.dremio.com"
]
}'
```

Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-by-distance"
  ],
  "createdAt": "2023-01-20T15:26:39.780Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "7cab1a42-8835-4d31-827b-fedee1ad38d1",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}
```

```

},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "other",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  }
]

```

```

    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Refresh the Reflections on a View

Refresh the Reflections associated with the specified view.

Read [Refreshing Reflections](#) to learn how refreshing works.

Method and URL

POST /api/v3/catalog/{id}/refresh

Parameters

id Path String (UUID)

Unique identifier for the view you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example

Request

```
curl -X POST 'https://api.dremio.cloud//api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Delete a View

Delete the specified view.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to delete.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[\[Previous](#)

[User-Defined Function\]\(/current/reference/api/catalog/user-defined-function\)\[Next](#)

[Lineage\]\(/current/reference/api/catalog/lineage\)](#)

[View Attributes](#)

[Create a View](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve a View by ID](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve a View by Path](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

-

Update a View

Parameters

Example

Response Status Codes

Refresh the Reflections on a View

Parameters

Example

Response Status Codes

Delete a View

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/lineage>

Version: current [26.x]

On this page

Lineage Enterprise

Use the Catalog API to retrieve lineage information about datasets (tables and views). The lineage object includes information about the dataset's sources, parent objects, and child objects.

Lineage Object

```
{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
      "createdAt": "2022-02-14T21:57:48.794Z"
    }
  ],
  "parents": [
    {
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
```

```

    "path": [
      "Samples",
      "samples.dremio.com",
      "zips.json"
    ],
    "tag": "MAntohVzwLw=",
    "type": "DATASET",
    "datasetType": "PROMOTED",
    "createdAt": "2023-01-18T18:49:09.669Z"
  },
  "children": [
    {
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
      "path": [
        "@dremio",
        "NYC_zip"
      ],
      "tag": "OWKrfpEKzW4=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T02:11:46.344Z"
    },
    {
      "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
      "path": [
        "@dremio",
        "Chicago_zip"
      ],
      "tag": "gsaDW5h4GCs=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T00:09:12.461Z"
    }
  ]
}

```

Lineage Attributes

sources Array of Object

Information about the sources the dataset uses. Each object in the sources array represents one source.

Example: [{"id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2","path": ["Samples"],"tag": "Iz1v71CeTQY=", "type": "CONTAINER", "containerType": "SOURCE", "createdAt": "2023-02-14T21:57:48.794Z"}]

parents Array of Object

Information about the parent objects for the dataset. Each object in the parents array represents one parent object. If a view represents a join of two other datasets, the

parents array includes the two joined datasets. The parents array is empty if the dataset does not have parent objects.

Example: [{"id": "3419fa3a-b5b3-4438-b864-a27ec4e18752", "path": ["Samples", "samples.dremio.com", "zips.json"], "tag": "MAntohVzwLw=", "type": "DATASET", "datasetType": "PROMOTED", "createdAt": "2023-01-18T18:49:09.669Z"}]

children Array of Object

Information about other catalog objects that reference the dataset. Each object in the children array represents one child object. The children array is empty if the dataset does not have child objects.

Example: [{"id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75", "path": ["@dremio", "NYC_zip"], "tag": "OWKrfpEKzW4=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-25T02:11:46.344Z"}, {"id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0", "path": ["@dremio", "Chicago_zip"], "tag": "gsaDW5h4GCs=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-25T00:09:12.461Z"}]

Attributes of Objects in the `sources` Array

id String (UUID)

Unique identifier of the source associated with the dataset.

Example: 21077e5d-fe6f-4a29-843f-58fa3acb17c2

path Array of String

Path of the source within Dremio, expressed as an array.

Example: ["Samples"]

tag String

Unique identifier of the version of the source. Dremio changes the tag whenever the source changes.

Example: lz1v71CeTQY=

type String

Type of source. For sources in lineage responses, the type is `CONTAINER`.

Example: CONTAINER

containerType String

Type of container for the source.

Enum: HOME, SOURCE

Example: SOURCE

createdAt String

Date and time that the source was created, in UTC format. Not included for sources with the containerType `HOME`.

Example: 2022-02-14T21:57:48.794Z

Attributes of Objects in the `parents` Array

id String (UUID)

Unique identifier of the parent object.

Example: 3419fa3a-b5b3-4438-b864-a27ec4e18752

path Array of String

Path of the parent object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the parent object itself as the last item in the array.

Example: ["Samples","samples.dremio.com","zips.json"]

tag String

Unique identifier of the version of the parent object. Dremio changes the tag whenever the parent object changes.

Example: MAntohVzwLw=

type String

Type of parent object. For parent objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the parent object. If the parent object is a table, `PROMOTED`. If the parent object is a view, `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: PROMOTED

createdAt String

Date and time that the parent object was created, in UTC format.

Example: 2023-01-18T18:49:09.669Z

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the child object.

Example: 170e211e-4235-4d8d-acb5-3d4dbfe99c75

path Array of String

Path of the child object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the child object itself as the last item in the array.

Example: ["@dremio","NYC_zip"]

tag String

Unique identifier of the version of the child object. Dremio changes the tag whenever the child object changes.

Example: OWKrfpEKzW4=

type String

Type of child object. For child objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the child object. For child objects in lineage responses, the datasetType is `VIRTUAL`.

Example: VIRTUAL

createdAt String

Date and time that the child object was created, in UTC format.

Example: 2023-01-25T02:11:46.344Z

Retrieve Lineage Information About a Dataset

Retrieve lineage information about the specified dataset.

Method and URL

```
GET /api/v3/catalog/{id}/graph
```

Parameters

id Path String (UUID)

Unique identifier of the dataset whose lineage you want to retrieve.

Example: d69b25a3-31c8-4d55-a7cc-dfee2290779b

Example

Request

```
curl -X GET
```

```
'https://{hostname}/api/v3/catalog/d69b25a3-31c8-4d55-a7cc-dfee2290779b/graph' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
{  
  "sources": [  
    {  
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",  
      "path": [  
        "Samples"  
      ],  
      "tag": "Iz1v71CeTQY=",  
      "type": "CONTAINER",  
      "containerType": "SOURCE",  
      "createdAt": "2022-02-14T21:57:48.794Z"  
    }  
  ],  
  "parents": [  
    {  
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",  
      "path": [  
        "Samples",  
        "samples.dremio.com",  
        "zips.json"  
      ],  
      "tag": "MAntohVzwLw=",  
      "type": "DATASET",  
      "datasetType": "PROMOTED",  
      "createdAt": "2023-01-18T18:49:09.669Z"  
    }  
  ],  
  "children": [  
    {  
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",  
      "path": [  
        "@dremio",  
        "NYC_zip"  
      ],  
      "tag": "0WKrfpEKzW4=",  
      "type": "DATASET",  
      "datasetType": "VIRTUAL",  
      "createdAt": "2023-01-25T02:11:46.344Z"  
    },  
    {  
      "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",  
      "path": [  
        "@dremio",  
        "Chicago_zip"  
      ],  
      "tag": "gsaDW5h4GCs=",  
      "type": "DATASET",
```

```
    "datasetType": "VIRTUAL",  
    "createdAt": "2023-01-25T00:09:12.461Z"  
  }  
]  
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

View](/current/reference/api/catalog/view)[Next

Tag](/current/reference/api/catalog/tag)

Lineage Attributes

Retrieve Lineage Information About a Dataset

Parameters

Example

Response Status Codes

Source:
[https://docs.dremio.com/current/reference/api/c
atalog/tag](https://docs.dremio.com/current/reference/api/catalog/tag)

Version: current [26.x]

On this page

Tag

Use the Catalog API to create, update, and retrieve tags.

Tag Object

```
{  
  "tags": [  
    "NYC",  
    "taxi",  
    "2023"
```

```
],  
  "version": "VJ3ijXH4m6k=",  
}
```

Tag Attributes

tags Array of String

List of tags that apply to the dataset.

Example: ["NYC","taxi","2023"]

version String

Unique identifier of the set of tags. Dremio changes the version whenever any of the tags change and uses the version value to ensure that updates apply to the most recent version of the set of tags.

Example: VJ3ijXH4m6k=

Create Tags

Create one or more tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset for which you want to add tags.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. Tags are case-insensitive. Each tag can be listed only once for each dataset. Each tag can have a maximum of 128 characters. Tags cannot include the following special characters: ```, `:`, `[`, or `]`.

Example: ["NYC","taxi","2023"]

Example

Request

```
curl -X POST  
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
```

```
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023"]
}'
```

Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve Tags

Retrieve the tags applied to the specified dataset.

Method and URL

```
GET /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Update Tags

Update the tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. If you want to keep any of the existing tags, include

them in the tags array. Tags are case-insensitive and must be distinct (in other words, list each tag only once for each dataset). Each tag may have a maximum of 128 characters. Tags cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: ["NYC","taxi","2023","archived"]

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are updating the most recent version of the tags.

Example: VM3ijXH4m6k=

Example

Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023", "archived"],
  "version": "VM3ijXH4m6k="
}'
```

Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023",
    "archived"
  ],
  "version": "yiZSE++9wiU="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Delete Tags

Delete the tags for the specified dataset.

note

Deleting tags means sending an empty array to replace the existing tags with no tags. The tag object will still exist, but it will contain an empty `tags` array and no tags will appear for the dataset in the Dremio UI.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to remove.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

Empty array to represent deletion of all tags for the dataset.

Example: []

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are deleting tags from the most recent version.

Example: yiZSE++9wiU=

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": [],
  "version": "yiZSE++9wiU="
}'
```

Response

```
{
  "tags": [],
```

```
"version": "wuTAKuRcVas="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[\[Previous](#)

[Lineage\]\(/current/reference/api/catalog/lineage\)\[Next](#)

[Wiki\]\(/current/reference/api/catalog/wiki\)](#)

[Tag Attributes](#)

[Create Tags](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve Tags](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Update Tags](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Delete Tags](#)

[Parameters](#)

[Example](#)

Source:
<https://docs.dremio.com/current/reference/api/catalog/wiki>

Version: current [26.x]

On this page

Wiki

Use the Catalog API to create, update, and retrieve the wiki for a source, space, or dataset.

Wiki Object

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio. Here is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",
  "version": 4
}
```

Wiki Attributes

text String

Text displayed in the wiki, formatted with GitHub-flavored Markdown.

version Integer

Number for the most recent version of the wiki, starting with `0`. Dremio increments the value by 1 each time the wiki changes and uses the version value to ensure that updates apply to the most recent version of the wiki.

Example: 4

Create a Wiki

Create a wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset for which you want to add the wiki.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki. Use [GitHub-flavored Markdown](#) for wiki formatting and ``\n`` for new lines and blank lines. Each wiki may have a maximum of 100,000 characters.

Example

Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum."
}'
```

Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
```

```

aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Wiki

Retrieve the wiki for the specified source, space, or dataset.

Method and URL

```
GET /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example

Request

```

curl -X GET
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wi
ki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'

```

Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.\nHere is some text in bold. Here is some text in italics.\n\nHere is an example\nexcerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur\nadipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna\naliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted\nlist\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###\nHeading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second\nitem in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that\nincludes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an\nimage:\n\n\n\nUt\nenim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea\ncommodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum\ndolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in\nculpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Update a Wiki

Update the wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki, formatted with GitHub-flavored Markdown.

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are updating the most recent version of the wiki.

Example: 0

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio. Here is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n## Heading Level 2\n\nHere is an update to the bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}'
```

Response

```
{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio. Here is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n## Heading Level 2\n\nHere is an update to the bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",
```

```
"version": 1
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Delete a Wiki

Delete the wiki for the specified source, space, or dataset.

note

Deleting the wiki entails sending an empty string to replace the existing wiki with no wiki. The wiki object will still exist, but it will contain an empty `text` value and no wiki will appear for the source, space, or dataset in the Dremio UI.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to delete.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Empty string to represent deletion of the wiki.

Example: ""

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are deleting the most recent version of the wiki.

Example: 1

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "",
  "version": 1
}'
```

Response

```
{
  "text": "",
  "version": 2
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[\[Previous](#)

[Tag\]/\(current/reference/api/catalog/tag\)\[Next](#)

[Grants\]/\(current/reference/api/catalog/grants\)](#)

[Wiki Attributes](#)

[Create a Wiki](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve a Wiki](#)

Parameters

Example

Response Status Codes

Update a Wiki

Parameters

Example

Response Status Codes

Delete a Wiki

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/grants>

Version: current [26.x]

On this page

Grants Enterprise

Use the Catalog API to grant user and role privileges on specific catalog objects.

Grants Object

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
```

```

    "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
    "name": "jeansmith",
    "firstName": "Jean",
    "lastName": "Smith",
    "email": "jean_smith@example.com"
  },
  {
    "privileges": [
      "ALTER",
      "SELECT"
    ],
    "granteeType": "ROLE",
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "name": "examplerole"
  }
]
}

```

Grants Attributes

id String

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

availablePrivileges Array of String

List of available privileges on the catalog object.

Example:

["ALTER","DELETE","INSERT","MANAGE\ GRANTS","SELECT","TRUNCATE","UPDATE"]

grants Array of Object

Information about the privileges and grantees for the catalog object. If the grants array is empty, there are no explicit grants for the object.

note

An empty grants array does not mean no users have access to the object at all. For example, admin users implicitly have all privileges on all catalog objects, owners implicitly have all privileges on everything they own, and children objects inherit the grants for their parent objects.

Example: [{"privileges": ["ALTER","SELECT","MANAGE\ GRANTS"],"granteeType": "USER","id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0","name": "jeansmith","firstName": "Jean","lastName": "Smith","email": "jean_smith@example.com"}, {"privileges": ["ALTER","SELECT"],"granteeType": "ROLE","id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","name": "examplerole"}]

Attributes of Objects in the `grants` Array

privileges String

List of privileges granted to the user or role. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE\GRANTS"]

granteeType String

Type of grantee.

Enum: USER, ROLE

Example: USER

id String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

name String

Name of the user or role.

Example: jeansmith

firstName String

For users, the user's first name. Not included for roles.

Example: Jean

lastName String

For users, the user's last name. Not included for roles.

Example: Smith

email String

For users, the user's email address. Not included for roles.

Example: jean_smith@example.com

Create or Updating Privilege Grants on a Catalog Object

Create or update the privileges granted to users and roles on the specified catalog object.

note

You must have the [MANAGE GRANTS privilege](#) to create or update privilege grants on catalog objects.

Method and URL

```
PUT /api/v3/catalog/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

grants Body Array of Object

Array of objects that specify which users and roles should have privileges on the catalog object, as well as each user's and role's specific privileges. May include objects for users, roles, or both.

Example: [{"privileges": ["ALTER","SELECT","MANAGE\ _GRANTS"],"granteeType": "USER","id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"}, {"privileges": ["SELECT","ALTER"],"granteeType": "ROLE","id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"}]

Parameters of Objects in the `grants` Array

privileges Body Array of String

List of privileges to grant to the user or role. Use the [Privileges](#) endpoint to retrieve a list of available privileges on the catalog object type. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE\ _GRANTS"]

granteeType Body String

Type of grantee.

Enum: USER, ROLE

Example: USER

id Body String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "grants": [
    {
      "privileges": [
```

```
    "ALTER",
    "SELECT",
    "MANAGE_GRANTS"
  ],
  "granteeType": "USER",
  "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"
},
{
  "privileges": [
    "SELECT",
    "ALTER"
  ],
  "granteeType": "ROLE",
  "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"
}
]
}'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve Privileges and Grantees on a Catalog Object

Retrieve information about the privileges granted to users and roles on the specified catalog object.

note

Use this endpoint in place of the Catalog API [Privileges](#) endpoint, which is deprecated. We expect to remove the Privileges endpoint by July 2025.

You must have the [MANAGE GRANTS privilege](#) to retrieve privilege grants on catalog objects.

Method and URL

GET /api/v3/catalog/{id}/grants

Parameters

id Path String (UUID)

Unique identifier of the object whose privilege grants you want to retrieve.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
      "firstName": "Jean",
      "lastName": "Smith",
      "email": "jean_smith@example.com"
    },
    {
      "privileges": [
        "ALTER",
        "SELECT"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "name": "examplerole"
    }
  ]
}
```

```
]
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Wiki](/current/reference/api/catalog/wiki)

Grants Attributes

Create or Updating Privilege Grants on a Catalog Object

Parameters

Example

Response Status Codes

Retrieve Privileges and Grantees on a Catalog Object

Parameters

Example

Response Status Codes

Source:
[https://docs.dremio.com/current/reference/api/c
atalog/container-space/](https://docs.dremio.com/current/reference/api/catalog/container-space/)

Version: current [26.x]

On this page

Space

Use the Catalog API to retrieve information about spaces and the child objects they contain, as well as to create, update, and delete spaces.

Space Object


```

{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
        "zips"
      ],
      "tag": "ITlp8+qyIMQ=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-08T16:24:25.084Z"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}

```

```

    ]
  }
]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",
  "TRUNCATE",
  "DELETE",
  "UPDATE",
  "EXECUTE",
  "CREATE_SOURCE",
  "ALL"
],
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Space Attributes

entityType String

Type of the catalog object. For spaces, the entityType is `space`.

Example: space

id String (UUID)

Unique identifier of the space.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

name String

Name of the space.

Example: Example-Space

tag String

Unique identifier of the version of the space. Dremio changes the tag whenever the space changes and uses the tag to ensure that PUT requests apply to the most recent

version of the space.

Example: zzOQfjY9IU0=

createdAt String

Date and time that the space was created, in UTC format.

Example: 2023-01-12T18:44:43.237Z

children Array of Object

Information about each catalog object in the space.

Example: [{"id": "8da037a1-8e50-422b-9a2b-cafb03f57c71", "path": ["Example-Space", "testfolder"], "tag": "0McuCL4MzBU=", "type": "CONTAINER", "containerType": "FOLDER"}, {"id": "f32df85-32e2-4c31-b2b4-bfd62ab3f473", "path": ["Example-Space", "travel_testing"], "tag": "i4mnlSmHqVM=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-30T17:54:25.547Z"}, {"id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda", "path": ["Example-Space", "zips"], "tag": "ITlp8+qyIMQ=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-02-08T16:24:25.084Z"}]

accessControlList Object

Enterprise only. Information about users and roles with access to the space and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if space-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the space. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ", "WRITE", "ALTER_REFLECTION", "SELECT", "ALTER", "VIEW_REFLECTION", "MODIFY", "MANAGE_GRANTS", "CREATE_TABLE", "DROP", "EXTERNAL_QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE_SOURCE", "ALL"]

owner Object

Information about the space's owner.

Example: {"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43", "ownerType": "USER"}

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the catalog object.

Example: 8da037a1-8e50-422b-9a2b-cafb03f57c71

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["Example-Space", "testfolder"]

tag String

Unique identifier of the version of the catalog object. Dremio changes the tag whenever the catalog object changes and uses the tag to ensure that PUT requests apply to the most recent version of the object.

Example: 0McuCL4MzBU=

type String

Type of the catalog object.

Enum: CONTAINER, DATASET, FILE

Example: CONTAINER

containerType String

For catalog entities with the type `CONTAINER`, the type of container.

Enum: FOLDER, FUNCTION

Example: FOLDER

datasetType String

For catalog objects in a space with the type `DATASET`, the datasetType is `VIRTUAL` (spaces cannot contain tables, only views).

Example: VIRTUAL

createdAt String

For catalog objects in a space with the type `DATASET`, date and time that the catalog object was created, in UTC format.

Example: 2023-01-30T17:54:25.547Z

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the space and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions":

```
["MODIFY"]}]
```

roles Array of Object

Enterprise only. List of roles whose members have access to the space and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the space's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the space.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise only. Unique identifier of the user or role with access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise only. List of privileges the user or role has on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["MODIFY"]

Create a Space

Create a new space.

Method and URL

```
POST /api/v3/catalog
```

Parameters

`entityType` Body String

Type of the catalog object. For spaces, the `entityType` is ``space``.

Example: `space`

`name` Body String

Name of the space. The name cannot include the following special characters: ``/`, `:`, `[`, or `]``.

Example: `Example-Space`

`accessControlList` Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure space-specific access control privileges.

Example: `{"users": [{ "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"] }], "roles": [{ "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"] }]}`

Parameters of the ``accessControlList`` Object

`users` Body Array of Object Optional

Enterprise only. List of users who should have access to the space and the specific privileges each user should have.

Example: `[{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5", "permissions": ["MODIFY"]}]`

`roles` Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have.

Example: `[{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]`

Parameters of Objects in the ``users`` and ``roles`` Arrays

`id` Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: `737a038f-c6cd-4fd3-a77a-59f692727ba5`

`permissions` Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["MODIFY"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "space",
  "name": "Example-Space",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}'
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  }
}
```

```

    ]
  }
],
"roles": [
  {
    "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
    "permissions": [
      "MODIFY"
    ]
  }
]
},
"permissions": [],
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve a Space by ID

Retrieve a space and information about its contents by specifying the space's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to retrieve.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdkyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
```

```

    "path": [
      "Example-Space",
      "testfolder"
    ],
    "tag": "0McuCL4MzBU=",
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
    "path": [
      "Example-Space",
      "travel_testing"
    ],
    "tag": "i4mnlSmHqVM=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-01-30T17:54:25.547Z"
  },
  {
    "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
    "path": [
      "Example-Space",
      "zips"
    ],
    "tag": "ITlp8+qyIMQ=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-08T16:24:25.084Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "MODIFY"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Space by Path

Retrieve a space and information about its contents by specifying the space's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the space that you want to retrieve. The path is the name of the space.

Example: Example-Space

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with

the `maxChildren` query parameter: the first request URL includes `maxChildren` set to the maximum number of child objects to include in each page of results. If the space has more child objects than the specified `maxChildren` value, the response includes a `nextPageToken` attribute. Add the `pageToken` query parameter with the `nextPageToken` value to the request URL to retrieve the next page of results. Do not remove or change the `maxChildren` query parameter when you add `pageToken` to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdkyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Example-Space' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mn1SmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
```

```

        "zips"
      ],
      "tag": "ITlp8+qyIMQ=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-08T16:24:25.084Z"
    }
  ],
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "MODIFY"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "MODIFY"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Response Status Codes

200 OK

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Update a Space

Update the specified space.

Method and URL

PUT /api/v3/catalog/{id}

Parameters

`id Path String (UUID)`

Unique identifier of the space that you want to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

`entityType Body String`

Type of the catalog object. For spaces, the `entityType` is ``space``.

Example: space

`id Body String (UUID)`

Unique identifier of the space to update.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

`name Body String`

Name of the space to update.

Example: Example-Space

`tag Body String`

Unique identifier of the version of the space that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the space.

Example: zzOQfjY9IU0=

`accessControlList Body String Optional`

Enterprise only. Object used to specify which users and roles should have access to the space and the specific privileges each user or role should have. If you omit the `accessControlList` object in a PUT request, Dremio removes all existing user and role access settings from the space. To keep existing user and role access settings while making other updates, duplicate the existing `accessControlList` array in the PUT request.

Example:

```
{
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": ["ALL"]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": ["MODIFY"]
    }
  ]
}
```

Parameters of the ``accessControlList`` Object

`users Body Array of Object Optional`

Enterprise only. List of users who should have access to the space and the specific privileges each user should have. If you omit the `users` object in a PUT request, Dremio removes all existing user access settings from the space. To keep existing user access settings while making other updates, duplicate the existing `users` array in the PUT request.

Example:

```
[
  {
    "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
    "permissions": ["ALL"]
  }
]
```

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the space and the specific privileges each role should have. If you omit the roles object in a PUT request, Dremio removes all existing role access settings from the space. To keep existing role access settings while making other updates, duplicate the existing roles array in the PUT request.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889", "permissions": ["MODIFY"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the space.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the space. For more information, read [Privileges](#).

Enum: ALL, VIEW_REFLECTION, TRUNCATE, UPDATE, DELETE, DROP, MANAGE_GRANTS, EXTERNAL_QUERY, EXECUTE, ALTER, INSERT, MODIFY, SELECT, CREATE_SOURCE, WRITE, CREATE_TABLE, ALTER_REFLECTION, READ

Example: ["ALL"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "zz0QfjY9lU0=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "ALL"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
```

```

        "permissions": [
            "MODIFY"
        ]
    }
}
}'

```

Response

```

{
  "entityType": "space",
  "id": "5442c00a-ada1-48c6-82fc-bb804b2e04e0",
  "name": "Example-Space",
  "tag": "PwZ6e/axHUY=",
  "createdAt": "2023-01-12T18:44:43.237Z",
  "children": [
    {
      "id": "8da037a1-8e50-422b-9a2b-cafb03f57c71",
      "path": [
        "Example-Space",
        "testfolder"
      ],
      "tag": "0McuCL4MzBU=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {
      "id": "f32dfe85-32e2-4c31-b2b4-bfd62ab3f473",
      "path": [
        "Example-Space",
        "travel_testing"
      ],
      "tag": "i4mnlSmHqVM=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-30T17:54:25.547Z"
    },
    {
      "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
      "path": [
        "Example-Space",
        "zips"
      ],
      "tag": "ITlp8+qyIMQ=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-02-08T16:24:25.084Z"
    }
  ],
  "accessControlList": {
    "users": [
      {

```



```

    "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
    "permissions": [
      "ALTER_REFLECTION",
      "MODIFY",
      "ALTER",
      "MANAGE_GRANTS",
      "VIEW_REFLECTION",
      "SELECT"
    ]
  },
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "MODIFY"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Delete a Space

Delete the specified space, including all of the space's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the space that you want to delete.

Example: 5442c00a-ada1-48c6-82fc-bb804b2e04e0

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Home]([current/reference/api/catalog/container-home])[Next

Folder]([current/reference/api/catalog/container-folder)

Space Attributes

Create a Space

Parameters

Example

Response Status Codes

Retrieve a Space by ID

Parameters

Example

Response Status Codes

Retrieve a Space by Path

Parameters

Example

Response Status Codes

Update a Space

Parameters

Example

Response Status Codes

Delete a Space

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/container-folder/>

Version: current [26.x]

On this page

Folder

Use the Catalog API to retrieve information about folders and the child objects they contain, as well as to create, update, and delete folders.

Folder Object

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
      ],
    }
  ],
}
```

```

    "type": "FILE"
  },
  {
    "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
    "path": [
      "Samples",
      "samples.dremio.com",
      "NYC-taxi-trips-iceberg"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
  },
  {
    "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
    "path": [
      "Samples",
      "samples.dremio.com",
      "NYC-taxi-trips"
    ],
    "type": "DATASET",
    "datasetType": "PROMOTED"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",

```

```

    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
  }
}

```

Folder Attributes

entityType String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

id String

Unique identifier of the folder. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

path Array of String

Path of the folder within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target folder itself as the last item in the array.

Example: ["Samples","samples.dremio.com"]

tag String

Unique identifier of the version of the folder. Dremio changes the tag whenever the folder changes and uses the tag to ensure that PUT requests apply to the most recent version of the folder.

Example: pRmJ0BQ9SFw=

children Array of Object

Information about each catalog object in the folder.

Example: [{"id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv","path": ["Samples","samples.dremio.com","zip_lookup.csv"],"type": "FILE"}, {"id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg","path": ["Samples","samples.dremio.com","NYC-taxi-trips-iceberg"],"type":

```
"CONTAINER","containerType": "FOLDER"}, {"id":
"1acab7b3-ee82-44c1-abcc-e86d56078d4d","path":
["Samples","samples.dremio.com","NYC-taxi-trips"],"type": "DATASET","datasetType":
"PROMOTED"}]
```

accessControlList Object

Enterprise-only. Information about users and roles with access to the folder and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if folder-specific access control privileges are not set.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the folder. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

```
["READ","WRITE","ALTER\_REFLECTION","SELECT","ALTER","VIEW\_REFLECTION","MODIFY","MANAGE\_GRANTS","CREATE\_TABLE","DROP","EXTERNAL\_QUERY","INSERT","TRUNCATE","DELETE","UPDATE","EXECUTE","CREATE\_SOURCE","ALL"]
```

owner Object

Information about the folder's owner.

Example: {"ownerId": "d01585a2-b267-4d56-9154-31762ab65a43","ownerType": "USER"}

Attributes of Objects in the `children` Array

id String

Unique identifier of the catalog object. The ID can be a UUID like `1acab7b3-ee82-44c1-abcc-e86d56078d4d` or a text path like `dremio:/Samples/samples.dremio.com/zip_lookup.csv`.

Example: dremio:/Samples/samples.dremio.com/zip_lookup.csv

path Array of String

Path of the catalog object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the catalog object itself as the last item in the array.

Example: ["Samples","samples.dremio.com","zip_lookup.csv"]

type String

Type of the catalog object. If the object is saved within a space (including the home space), valid types are `CONTAINER` and `DATASET`. If the object is saved within a

source, valid types are `CONTAINER`, `FILE`, and `DATASET`.

Example: CONTAINER

containerType String

For catalog objects with the type `CONTAINER`, the containerType is `FOLDER`.

Example: FOLDER

datasetType String

For catalog objects with the type `DATASET`, the type of dataset. For tables, the datasetType is `PROMOTED`. For views, the datasetType is `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: VIRTUAL

createdAt String

Date and time that the catalog object was created, in UTC format. The createdAt attribute is included only for `DATASET` catalog objects that are saved in folders within spaces, not within sources.

Example: 2023-01-30T17:54:25.547Z

Attributes of the `accessControlList` Object

users String

Enterprise-only. List of users with access to the folder and the specific privileges each user has.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions":["SELECT","ALTER"]}]

roles String

Enterprise-only. List of roles whose members have access to the folder and the specific privileges each role has.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions":["SELECT"]}]

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the folder's owner.

Example: d01585a2-b267-4d56-9154-31762ab65a43

ownerType String

Type of owner of the folder.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `users` and `roles` Arrays

id String (UUID)

Enterprise-only. Unique identifier of the user or role with access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Array of String

Enterprise-only. List of privileges the user or role has on the folder. For more information, read [Privileges](#).

Example: ["SELECT","ALTER"]

Create a Folder

Create a new folder within a space.

note

The Catalog API cannot create new folders within sources.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For folders, the entityType is `folder`.

Example: folder

path Body Array of String

Path of the location where the folder should be created within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the target folder itself as the last item in the array. The name of the folder cannot include the following special characters: `/`, `:`, `[`, or `.`.

Example: ["Example-Space","First-Folder","New-Folder"]

[accessControlList](#) Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. May include an array of users, an array of roles, or both. Omit if you do not want to configure folder-specific access control privileges.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}], "roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["SELECT","ALTER"]}]

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["SELECT","ALTER"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "folder",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ]
}
```

```

],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
}
}'

```

Response

```

{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "R7C0ubQq8KE=",
  "accessControlList": {
    "users": [
      {
        "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ],
    "roles": [
      {
        "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
        "permissions": [
          "SELECT"
        ]
      }
    ]
  },
  "permissions": [],
}

```

```
"owner": {  
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",  
  "ownerType": "USER"  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve a Folder by ID

Retrieve a folder and information about its contents by specifying the folder's ID.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to retrieve. If the ID is a text path, use URL encoding to replace any special characters with their UTF-8-equivalent characters, such as ``%3A`` for a colon; ``%2F`` for a forward slash; and ``%20`` for a space. For example, if the ID value is ``dremio:/Samples/samples.dremio.com/Dremio University``, the URI-encoded ID is ``dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University``.

Example: d4c2a8ba-a972-4db4-8deb-67e1ade684d1

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is ``permissions``. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query

parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for folders in filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/d4c2a8ba-a972-4db4-8deb-67e1ade684d1' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
```

```

        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
    ],
    "type": "FILE"
},
{
    "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
    "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips-iceberg"
    ],
    "type": "CONTAINER",
    "containerType": "FOLDER"
},
{
    "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
    "path": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
    ],
    "type": "DATASET",
    "datasetType": "PROMOTED"
}
],
"accessControlList": {
    "users": [
        {
            "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
            "permissions": [
                "SELECT",
                "ALTER"
            ]
        }
    ],
    "roles": [
        {
            "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
            "permissions": [
                "SELECT"
            ]
        }
    ]
},
"owner": {
    "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
    "ownerType": "USER"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Folder by Path

Retrieve a folder and information about its contents by specifying the folder's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the folder that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Samples/samples.dremio.com

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

exclude Query String Optional

Exclude a default attribute from the response. The available value for the exclude query parameter is `children`. For more information, read [include and exclude Query Parameters](#).

Example: ?exclude=children

maxChildren Query Integer Optional

Specify the maximum number of child objects to include in each page of results. Use in concert with the pageToken query parameter to split large sets of results into multiple pages. For more information, read [maxChildren Query Parameter](#).

NOTE: The maxChildren query parameter is not supported for filesystem sources.

Example: ?maxChildren=25

pageToken Query String Optional

Specify the token for retrieving the next page of results. Must be used in concert with the maxChildren query parameter: the first request URL includes maxChildren set to the maximum number of child objects to include in each page of results. If the folder has more child objects than the specified maxChildren value, the response includes a nextPageToken attribute. Add the pageToken query parameter with the nextPageToken value to the request URL to retrieve the next page of results. Do not remove or change the maxChildren query parameter when you add pageToken to the request URL. Read [pageToken Query Parameter: User-Specified Maximum](#) for more information.

NOTE: Dremio ignores the pageToken query parameter for folders in filesystem sources.

Example:

?pageToken=cHAAFLceQCKsTVpwaEVisqgjDntZJUCuTqVNghPdKyBDUNoJvwrEXAMPLE

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "folder",
  "id": "d4c2a8ba-a972-4db4-8deb-67e1ade684d1",
  "path": [
    "Samples",
    "samples.dremio.com"
  ],
  "tag": "pRmJ0BQ9SFw=",
  "children": [
    {
      "id": "dremio:/Samples/samples.dremio.com/zip_lookup.csv",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zip_lookup.csv"
      ],
      "type": "FILE"
    },
    {
      "id": "dremio:/Samples/samples.dremio.com/NYC-taxi-trips-iceberg",
      "path": [
        "Samples",
        "samples.dremio.com",

```

```

    "NYC-taxi-trips-iceberg"
  ],
  "type": "CONTAINER",
  "containerType": "FOLDER"
},
{
  "id": "1acab7b3-ee82-44c1-abcc-e86d56078d4d",
  "path": [
    "Samples",
    "samples.dremio.com",
    "NYC-taxi-trips"
  ],
  "type": "DATASET",
  "datasetType": "PROMOTED"
}
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Update a Folder

Update the specified folder.

Method and URL

```
PUT /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

entityType Body String

Type of the catalog object. For folders, the entityType is ``folder``.

Example: folder

id Body String

Unique identifier of the folder to update. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

path Body Array of String

Path of the location where the folder is saved within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the target folder itself as the last item in the array.

Example: ["Example-Space", "First-Folder", "New-Folder"]

tag Body String

Unique identifier of the version of the folder that you want to update. Dremio uses the tag to ensure that you are requesting to update the most recent version of the folder.

Example: R7COubQq8KE=

accessControlList Body Object Optional

Enterprise-only. Object used to specify which users and roles should have access to the folder and the specific privileges each user or role should have. If you omit the accessControlList object in a PUT request, Dremio removes all existing user and role access settings from the folder. To keep existing user and role access settings while making other updates, duplicate the existing accessControlList array in the PUT request.

Example: {"users": [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALL"]}],"roles": [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]}

Parameters of the `accessControlList` Object

users Body String Optional

Enterprise-only. List of users who should have access to the folder and the specific privileges each user should have. If you omit the users object in a PUT request, Dremio removes all existing user access settings from the folder. To keep existing user access settings while making other updates, duplicate the existing users array in the PUT request.

Example: [{"id": "737a038f-c6cd-4fd3-a77a-59f692727ba5","permissions": ["ALL"]}]

roles Body String Optional

Enterprise-only. List of roles whose members should have access to the folder and the specific privileges each role should have. If you omit the roles object in a PUT request, Dremio removes all existing role access settings from the folder. To keep existing role access settings while making other updates, duplicate the existing roles array in the PUT request.

Example: [{"id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","permissions": ["SELECT"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String (UUID) Optional

Enterprise-only. Unique identifier of the user or role who should have access to the folder.

Example: 737a038f-c6cd-4fd3-a77a-59f692727ba5

permissions Body Array of String Optional

Enterprise-only. List of privileges the user or role should have on the folder. For more information, read [Privileges](#).

Enum: ALTER_REFLECTION, SELECT, ALTER, VIEW_REFLECTION, MANAGE_GRANTS, ALL

Example: ["ALL"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/5442c00a-ada1-48c6-82fc-bb804b2e04e0' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
```

```

"entityType": "folder",
"id": "598697c2-8be0-4050-9731-53563977a17d",
"path": [
  "Example-Space",
  "First-Folder",
  "New-Folder"
],
>tag": "R7C0ubQq8KE=",
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALL"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
}
}'

```

Response

```

{
  "entityType": "folder",
  "id": "598697c2-8be0-4050-9731-53563977a17d",
  "path": [
    "Example-Space",
    "First-Folder",
    "New-Folder"
  ],
  "tag": "vnRnYLLpCFU=",
  "children": [
    {
      "id": "d60f9258-e55a-4fc3-97b3-58c6720a70fc",
      "path": [
        "Example-Space",
        "First-Folder",
        "New-Folder",
        "NYC-trips-weather"
      ],
      "tag": "IHXU70xs80c=",
      "type": "CONTAINER",
      "containerType": "FOLDER"
    },
    {

```

```

    "id": "acba8595-bfcf-4126-887c-d2a19b5afb1d",
    "path": [
      "Example-Space",
      "First-Folder",
      "New-Folder",
      "short-distance-trips"
    ],
    "tag": "KYs/Qyw1ok8=",
    "type": "DATASET",
    "datasetType": "VIRTUAL",
    "createdAt": "2023-02-09T19:09:58.789Z"
  }
],
"accessControlList": {
  "users": [
    {
      "id": "737a038f-c6cd-4fd3-a77a-59f692727ba5",
      "permissions": [
        "ALTER_REFLECTION",
        "ALTER",
        "MANAGE_GRANTS",
        "VIEW_REFLECTION",
        "SELECT"
      ]
    }
  ],
  "roles": [
    {
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "permissions": [
        "SELECT"
      ]
    }
  ]
},
"owner": {
  "ownerId": "d01585a2-b267-4d56-9154-31762ab65a43",
  "ownerType": "USER"
}
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Delete a Folder

Delete the specified folder, including all of the folder's contents.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the folder that you want to delete. The ID can be a UUID like ``1acab7b3-ee82-44c1-abcc-e86d56078d4d`` or a text path like ``dremio:/Samples/samples.dremio.com/zip_lookup.csv``.

Example: 598697c2-8be0-4050-9731-53563977a17d

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/598697c2-8be0-4050-9731-53563977a17d' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Space]([current/reference/api/catalog/container-space])[Next

File]([current/reference/api/catalog/file])

Folder Attributes

Create a Folder

Parameters

Example

Response Status Codes

Retrieve a Folder by ID

Parameters

Example

Response Status Codes

Retrieve a Folder by Path

Parameters

Example

Response Status Codes

Update a Folder

Parameters

Example

Response Status Codes

Delete a Folder

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/file/>

Version: current [26.x]

On this page

File

Use the Catalog API to retrieve information about formatting data to a table.

File Object

```
{
```

```
"entityType": "file",
"id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
"path": [
  "Samples",
  "samples.dremio.com",
  "SF weather 2018-2019.csv"
]
}
```

File Attributes

entityType String

Type of the catalog object. For files, the entityType is `file`.

Example: file

id String

Unique identifier of the file. For files, the ID is the text path of the file within Dremio.

Example: dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv

path Array of String

Path of the file within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the target file itself as the last item in the array.

Example: ["Samples","samples.dremio.com","SF weather 2018-2019.csv"]

Retrieve a File by Path

Retrieve information about a file by specifying its path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Path of the file that you want to retrieve, with a forward slash to separate each level of nesting. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, replace colons with `%3A` and replace spaces with `%20`.

Example: Samples/samples.dremio.com/SF%20weather%202018-2019.csv

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/SF%20weather%2018-2019.csv' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "file",
  "id": "dremio:/Samples/samples.dremio.com/SF weather 2018-2019.csv",
  "path": [
    "Samples",
    "samples.dremio.com",
    "SF weather 2018-2019.csv"
  ]
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

500 Internal Server Error

Was this page helpful?

[Previous

Folder]([current/reference/api/catalog/container-folder])[Next

Table]([current/reference/api/catalog/table)

File Attributes

Retrieve a File by Path

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/table/>

Version: current [26.x]

On this page

Table

Use the Catalog API to retrieve tables, format files and folders as tables, update and refresh tables, and revert tables to files and folders.

Table Object

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "SCHEDULE",
    "refreshPeriodMs": 3600000,
    "gracePeriodMs": 10800000,
    "refreshSchedule": "0 0 8 * * ?",
    "method": "FULL",
    "neverExpire": false,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": false,
    "autoCorrectCorruptDates": true
  },
}
```

```

"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"permissions": [
  "READ",
  "WRITE",
  "ALTER_REFLECTION",
  "SELECT",
  "ALTER",
  "VIEW_REFLECTION",
  "MODIFY",
  "MANAGE_GRANTS",
  "CREATE_TABLE",
  "DROP",
  "EXTERNAL_QUERY",
  "INSERT",
  "TRUNCATE",
  "DELETE",
  "UPDATE",
  "EXECUTE",
  "CREATE_SOURCE",
  "ALL"
],
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {

```

```

    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [
        {
          "type": {
            "name": "VARCHAR"
          }
        }
      ]
    }
  },
  {
    "name": "review_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "stars",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "attributes",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {

```

```

    "name": "Parking",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "garage",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "street",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "lot",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "valet",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },
  {
    "name": "Accepts Credit Cards",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Wheelchair Accessible",
    "type": {
      "name": "BOOLEAN"
    }
  },
  {
    "name": "Price Range",
    "type": {
      "name": "BIGINT"
    }
  }
]
},
{
  "name": "date",

```

```

    "type": {
      "name": "VARCHAR"
    }
  },
  "approximateStatisticsAllowed": false
}

```

Table Attributes

entityType String

Type of the catalog object. For tables, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the table.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

type String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example: PHYSICAL_DATASET

path Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples","samples.dremio.com","Dremio University","restaurant_reviews.parquet"]

createdAt String

Date and time that the table was created, in UTC format.

Example: 2024-01-13T19:52:01.894Z

tag String (UUID)

Unique identifier of the version of the table. Dremio changes the tag whenever the table changes and uses the tag to ensure that PUT requests apply to the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

accelerationRefreshPolicy String

Attributes that define the acceleration refresh policy for the table.

isMetadataExpired Boolean

If true, the metadata of the table needs to be refreshed. To refresh it, run the ALTER TABLE command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the table.

If NULL, metadata has never yet been collected for the table.

`lastMetadataRefreshAt` String

Date and time that the table metadata was last refreshed. In UTC format. If NULL, the metadata has never yet been refreshed.

Example: 2024-01-31T09:50:01.012Z

format Object

Table format attributes.

accessControlList Object

Enterprise only. Information about users and roles with access to the table and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if table-specific access control privileges are not set.

Example: `{"users": [{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"] }, { "id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\ _GRANTS"] }], "roles": [{ "id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"] }] }`

`permissions` Array of String

Enterprise-only. List of the privileges that you have on the table. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

`["READ", "WRITE", "ALTER\ _REFLECTION", "SELECT", "ALTER", "VIEW\ _REFLECTION", "MODIFY", "MANAGE\ _GRANTS", "CREATE\ _TABLE", "DROP", "EXTERNAL\ _QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE\ _SOURCE", "ALL"]`

owner String

Information about the table's owner.

fields Object

Attributes that represent the table schema.

`approximateStatisticsAllowed` Boolean

If true, `COUNT DISTINCT` queries run on the table return approximate results. Otherwise, false.

Example: `{ "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8", "ownerType": "USER" }`

Attributes of the `accelerationRefreshPolicy` Object

activePolicyType String

Option to set the policy for refreshing Reflections that are defined on the source. For this option to take effect, `neverRefresh` must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

`PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by refreshPeriodMs.

`SCHEDULE`: The Reflections are refreshed according to the schedule that is set by refreshSchedule.

`REFRESH_ON_DATA_CHANGES`: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency.

refreshPeriodMs Integer

Refresh period for the data in all Reflections for the table, in milliseconds.

Example: 3600000

refreshSchedule String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source are refreshed.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

Special Character	Description
---	---
*	Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week.
?	Equivalent to *.
,	Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.
-	Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Integer

Maximum age allowed for Reflection data used to accelerate queries, in milliseconds.

Example: 10800000

method String

Approach used for refreshing the data in Reflections defined on tables that are not in the Apache Iceberg format. For more information, read [Refreshing Reflections](#).

Enum: FULL, INCREMENTAL

Example: FULL

refreshField String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Boolean

If the Reflection never expires, the value is `true`. Otherwise, the value is `false`.

Example: false

neverRefresh Boolean

If the Reflection never refreshes, the value is `true`. Otherwise, the value is `false`.

Example: false

sourceRefreshOnDataChanges Boolean

If the table's source is configured so that Reflections on tables in Iceberg format in the source will refresh when new snapshots are created after an update, `true`. Otherwise, `false`.

Attributes of the `format` Object

type String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

Example: Parquet

name String

Table name. Dremio automatically duplicates the name of the origin file or folder to

populate this value. The name of the origin file or folder cannot include the following special characters: `/`, `:`, `[`, or `]`.

Example: restaurant_reviews.parquet

fullPath Array of String

Path of the table within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

ctime Integer

Not used. Has the value `0`.

Example: 0

isFolder Boolean

If the value is `true`, the table was created from a folder. If the value is `false`, the table was created from a file.

Example: false

location String

Location, expressed as a string, where the table's metadata is stored within a Dremio source or space.

Example: /samples.dremio.com/Dremio University/restaurant_reviews.parquet

ignoreOtherFileFormats Boolean

If true, Dremio ignores all non-Parquet files in the related folder structure, and the promoted table works as if only Parquet files are in the folder structure. Otherwise, false. Included only for Parquet folders.

Example: false

metaStoreType String

Not used. Has the value `HDFS`.

Example: HDFS

parquetDataFormat Object

Information about data format for Parquet tables.

dataFormatTypeList Array of String

List of data format types in the table. Included only for Iceberg tables, and `PARQUET` is the only valid value.

Example: ["PARQUET"]

sheetName String

For tables created from files that contain multiple sheets, the name of the sheet used to create the table.

Example: location_1

extractHeader Boolean

For tables created from files, the value is ``true`` if Dremio extracted the table's column names from the first line of the file. Otherwise, the value is ``false``.

Example: false

hasMergedCells Boolean

For tables created from files, the value is ``true`` if Dremio expanded merged cells in the file when creating the table. Otherwise, the value is ``false``.

Example: true

fieldDelimiter String

Character used to indicate separate fields in the table. May be ```,`` for a comma (default), ``\t`` for a tab, ``|`` for a pipe, or a custom character.

quote String

Character used for quotation marks in the table. May be ``\"`` for a double quote (default), ``'`` for a single quote, or a custom character.

comment String

Character used to indicate comments in the table. May be ``#`` for a number sign (default) or a custom character.

escape String

Character used to indicate an escape in the table. May be ``\"`` for a double quote (default), ``\`` for a back quote, ``\\`` for a backward slash, or a custom character.

lineDelimiter String

Character used to indicate separate lines in the table. May be ``\r\n`` for a carriage return and a new line (default), ``\n`` for a new line, or a custom character.

skipFirstLine Boolean

If Dremio skipped the first line in the file or folder when creating the table, the value is ``true``. Otherwise, the value is ``false``.

Example: false

autoGenerateColumnNames Boolean

If Dremio used the existing column names in the file or folder for the table columns, the value is ``true``. Otherwise, the value is ``false``.

Example: true

trimHeader Boolean

If Dremio trimmed column names to a specific number of characters when creating the table, the value is ``true``. Otherwise, the value is ``false``.

Example: true

autoCorrectCorruptDates Boolean

If Dremio automatically corrects corrupted date fields in the table, the value is ``true``. Otherwise, the value is ``false``.

Example: true

Attributes of the ``accessControlList`` Object

users Array of Object

Enterprise only. List of users with access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]

roles Array of Object

Enterprise only. List of roles whose members have access to the table and the specific privileges each role has.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT", "ALTER"]}]

Attributes of Objects in the ``users`` and ``roles`` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Array of String

Enterprise only. List of privileges the user or role has on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Attributes of the ``owner`` Object

ownerId String (UUID)

Unique identifier of the table's owner.

Example: 30fca499-4abc-4469-7142-fc8dd29acac8

ownerType String

Type of owner of the table.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the table field.

Example: review_count

type Object

Information about the table field.

Attributes of the `type` Object

name String

Name of the table field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: BIGINT

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 0

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema appears only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: Parking

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Attributes of the `parquetDataFormat` Object

type String

Type of data in the table. Within the parquetDataFormat object, the only valid type is `Parquet`.

Example: Parquet

ctime Integer

Not used. Has the value `0`.

Example: 0

isFolder Boolean

If the value is `true`, the table was created from a folder. If the value is `false`, the table was created from a file.

Example: true

autoCorrectCorruptDates Boolean

If the value is `true`, Dremio automatically corrects corrupted date fields in the table. Otherwise, the value is `false`.

Example: true

Format a File or Folder as a Table

Format a file or folder as a table so that you can query the data in Dremio.

note

To format a folder, all files in the folder must be the same format.

Method and URL

```
POST /api/v3/catalog/{id}
```

Parameters

id Path String

Unique identifier of the file or folder you want to format. The ID can be a UUID or a text path. If the ID is a text path, use URL encoding to replace special characters with their UTF-8-equivalent characters: `%3A` for a colon; `%2F` for a forward slash, and `%20` for a space. For example, if the ID value is `dremio:/Samples/samples.dremio.com/Dremio University`, the URI-encoded ID is `dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University`.

Example: c590ed7f-7142-4e1f-ba7d-94173afdc9a3

entityType Body String

Type of the catalog object. To format a file or folder as a table, the entityType is `dataset`.

path Body Array of String

Path of the file or folder you want to format, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the file or folder itself as the last item in the array. Get the path from the file or folder's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

type Body String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

[format](#) Body String

Parameters that describe how to format the file or folder.

[accessControllist](#) Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

Parameters of the `accelerationRefreshPolicy` Object

activePolicyType Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the neverRefresh parameter must be set to `false`.

The possible values are:

`NEVER`: The Reflections are never refreshed.

``PERIOD``: Default. The Reflections are refreshed at the end of every period that is defined by `refreshPeriodMs`.

``SCHEDULE``: The Reflections are refreshed according to the schedule that is set by `refreshSchedule`.

``REFRESH_ON_DATA_CHANGES``: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in Iceberg format.

`refreshPeriodMs` Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set `activePolicyType` to ``PERIOD``. The default setting is ``3600000`` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

`refreshSchedule` Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set `activePolicyType` to ``SCHEDULE``. The default `refreshSchedule` setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0	N/A
Minute	0-59	N/A
Hour	0-23	N/A
Day of month	N/A	* ?
Month	N/A	* ?
Days of week	1-7 or SUN-SAT	, - * ?

Special Character	Description
---	---
*	Used to specify all values for a field. For <code>`Day of month`</code> , specifies every day of the month. For <code>`Month`</code> , specifies every month. For <code>`Days of week`</code> , specifies every day of the week.
?	Equivalent to <code>*</code> .
,	Used to specify two or more days in the <code>`Days of week`</code> field. For example, <code>`MON,WED,FRI`</code> .
-	Used to specify ranges in the <code>`Days of week`</code> field. For example, <code>`1-3`</code> is equivalent to <code>`Sunday, Monday, and Tuesday`</code> .

Examples:

``0 0 0 * * ?`` : Refreshes every day at midnight.

``0 45 15 * * 1,4,7`` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

``0 15 7 ? * 2-6`` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in AWS Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the file or folder. All files in the folder must be the same format.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

ignoreOtherFileFormats Body Boolean Optional

If Dremio should ignore all non-Parquet files in the related folder structure so that the promoted table works as if only Parquet files are in the folder structure, set to `true`. Otherwise, set to `false` (default). Optional for Parquet folders.

Example: false

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the file or folder when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Excel and Text types.

Example: true

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the file when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Excel type.

Example: true

sheetName Body String Optional

For tables created from Excel files that contain multiple sheets, the name of the sheet to use to create the table. Default is the first sheet in the file (for files that contain multiple sheets).

Example: location_1

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character. Optional for files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be `"` for a double quote (default), `'` for a single quote, or a custom character. Optional for files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments in the table. May be `#` for a number sign (default) or a custom character. Optional for files or folders of the Text type.

escapeBody String Optional

Character used to indicate an escape in the table. May be `"` for a double quote (default), `` ` `` for a back quote, ` \ ` for a backward slash, or a custom character. Optional for files or folders of the Text type.

lineDelimiter String Optional

Character used to indicate separate lines in the table. May be ` \n ` for a carriage return and a new line (default), ` \n ` for a new line, or a custom character. Optional for files or folders of the Text type.

autoGenerateColumnNames Boolean Optional

If Dremio should use the existing column names in the file or folder for the table columns, set to `true` (default). Otherwise, set to `false`. Optional for files or folders of the Text type.

Example: true

trimHeader Boolean Optional

If Dremio should trim column names to a specific number of characters when creating the table, set to `true`. Otherwise, set to `false` (default). Optional for files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Samples",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Parquet"
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}
```

```
}'
```

Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "format": {
    "type": "Parquet",
    "name": "restaurant_reviews.parquet",
    "fullPath": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "restaurant_reviews.parquet"
    ],
    "ctime": 0,
    "isFolder": false,
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
    "ignoreOtherFileFormats": "false",
    "autoCorrectCorruptDates": true
  },
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  }
}
```

```

        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
            "SELECT",
            "ALTER",
            "MANAGE_GRANTS"
        ]
    },
],
"roles": [
    {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
            "SELECT",
            "ALTER"
        ]
    }
],
},
"owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
},
"fields": [
    {
        "name": "_id",
        "type": {
            "name": "VARCHAR"
        }
    },
    {
        "name": "name",
        "type": {
            "name": "VARCHAR"
        }
    },
    {
        "name": "city",
        "type": {
            "name": "VARCHAR"
        }
    },
    {
        "name": "state",
        "type": {
            "name": "VARCHAR"
        }
    },
    {
        "name": "categories",
        "type": {
            "name": "LIST",
            "subSchema": [
                {
                    "type": {

```

```

        "name": "VARCHAR"
    }
}
]
}
},
{
    "name": "review_count",
    "type": {
        "name": "BIGINT"
    }
},
{
    "name": "stars",
    "type": {
        "name": "DOUBLE"
    }
},
{
    "name": "attributes",
    "type": {
        "name": "STRUCT",
        "subSchema": [
            {
                "name": "Parking",
                "type": {
                    "name": "STRUCT",
                    "subSchema": [
                        {
                            "name": "garage",
                            "type": {
                                "name": "BOOLEAN"
                            }
                        }
                    ],
                }
            },
            {
                "name": "street",
                "type": {
                    "name": "BOOLEAN"
                }
            },
            {
                "name": "lot",
                "type": {
                    "name": "BOOLEAN"
                }
            },
            {
                "name": "valet",
                "type": {
                    "name": "BOOLEAN"
                }
            }
        ]
    }
}
]
}

```

```

    },
    {
      "name": "Accepts Credit Cards",
      "type": {
        "name": "BOOLEAN"
      }
    },
    {
      "name": "Wheelchair Accessible",
      "type": {
        "name": "BOOLEAN"
      }
    },
    {
      "name": "Price Range",
      "type": {
        "name": "BIGINT"
      }
    }
  ]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Example Request for Excel format type

```

curl -X POST \
  'https://{hostname}/api/v3/catalog/dremio%3A%2FSamples%2Fsamples.dremio.com%2FDremio%20University%2Foracle-departments.xlsx' \
  --header 'Authorization: Bearer <dremioAccessToken>' \
  --header 'Content-Type: application/json' \
  --data-raw '{
    "entityType": "dataset",
    "path": [
      "Samples",
      "samples.dremio.com",
      "Dremio University",
      "oracle-departments.xlsx"
    ],
    "type": "PHYSICAL_DATASET",
    "format": {
      "type": "Excel",
      "extractHeader": true,
      "hasMergedCells": true,
      "sheetName": "Sheet1"
    }
  }'

```

```
}  
}'
```

Example Request for Text format type

```
curl -X POST 'https://{hostname}/api/v3/catalog/6ba3bd6e-fd27-4572-a535-77e1548283b3' \  
  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json' \  
--data-raw '{  
  "entityType": "dataset",  
  "path": [  
    "Samples",  
    "samples.dremio.com",  
    "Dremio University",  
    "airbnb_listings.csv"  
  ],  
  "type": "PHYSICAL_DATASET",  
  "format": {  
    "type": "Text",  
    "fieldDelimiter": ",",  
    "skipFirstLine": false,  
    "extractHeader": true,  
    "quote": "\"",  
    "comment": "#",  
    "escape": "\\",  
    "lineDelimiter": "\r\n",  
    "autoGenerateColumnNames": true,  
    "trimHeader": false  
  }  
}'
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a Table by ID

Retrieve a table by specifying the table's `id` value.

Method and URL

Parameters

id Path String (UUID)

Unique identifier of the table that you want to retrieve.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "dataset",
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "createdAt": "2024-01-13T19:52:01.894Z",
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",
  "accelerationRefreshPolicy": {
    "activePolicyType": "PERIOD",
    "refreshPeriodMs": 3600000,
    "refreshSchedule": "0 56 18 * * *",
    "gracePeriodMs": 259200000,
    "method": "FULL",
    "neverExpire": true,
    "neverRefresh": false,
    "sourceRefreshOnDataChanges": false
  },
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
```

```

"format": {
  "type": "Parquet",
  "name": "restaurant_reviews.parquet",
  "fullPath": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "restaurant_reviews.parquet"
  ],
  "ctime": 0,
  "isFolder": false,
  "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",
  "ignoreOtherFileFormats": false,
  "autoCorrectCorruptDates": true
},
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  }
],

```

```

{
  "name": "name",
  "type": {
    "name": "VARCHAR"
  }
},
{
  "name": "city",
  "type": {
    "name": "VARCHAR"
  }
},
{
  "name": "state",
  "type": {
    "name": "VARCHAR"
  }
},
{
  "name": "categories",
  "type": {
    "name": "LIST",
    "subSchema": [
      {
        "type": {
          "name": "VARCHAR"
        }
      }
    ]
  }
},
{
  "name": "review_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "stars",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "attributes",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "Parking",
        "type": {
          "name": "STRUCT",
          "subSchema": [
            {

```

```

        "name": "garage",
        "type": {
            "name": "BOOLEAN"
        }
    },
    {
        "name": "street",
        "type": {
            "name": "BOOLEAN"
        }
    },
    {
        "name": "lot",
        "type": {
            "name": "BOOLEAN"
        }
    },
    {
        "name": "valet",
        "type": {
            "name": "BOOLEAN"
        }
    }
]
}
},
{
    "name": "Accepts Credit Cards",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Wheelchair Accessible",
    "type": {
        "name": "BOOLEAN"
    }
},
{
    "name": "Price Range",
    "type": {
        "name": "BIGINT"
    }
}
]
}
},
{
    "name": "date",
    "type": {
        "name": "VARCHAR"
    }
}
],

```

```
"approximateStatisticsAllowed": false
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Table by Path

Retrieve a table by specifying the table's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

Table's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" table in the "samples.dremio.com" folder within the source "Samples," the path is `Samples/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example:

Samples/samples.dremio.com/Dremio%20University/restaurant_reviews.parquet

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Samples/samples.dremio.com/Dremio%20Universit
```

```
y/restaurant_reviews.parquet' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
{  
  "entityType": "dataset",  
  "id": "c9c11d32-0576-4200-5a5b-8c7229cb3d72",  
  "type": "PHYSICAL_DATASET",  
  "path": [  
    "Samples",  
    "samples.dremio.com",  
    "Dremio University",  
    "restaurant_reviews.parquet"  
  ],  
  "createdAt": "2024-01-13T19:52:01.894Z",  
  "tag": "cb2905bb-39c0-497f-ae74-4c310d534f25",  
  "accelerationRefreshPolicy": {  
    "activePolicyType": "PERIOD",  
    "refreshPeriodMs": 3600000,  
    "refreshSchedule": "0 56 18 * * *",  
    "gracePeriodMs": 259200000,  
    "method": "FULL",  
    "neverExpire": true,  
    "neverRefresh": false,  
    "sourceRefreshOnDataChanges": false  
  },  
  "isMetadataExpired": false,  
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",  
  "format": {  
    "type": "Parquet",  
    "name": "restaurant_reviews.parquet",  
    "fullPath": [  
      "Samples",  
      "samples.dremio.com",  
      "Dremio University",  
      "restaurant_reviews.parquet"  
    ],  
    "ctime": 0,  
    "isFolder": false,  
    "location": "/samples.dremio.com/Dremio University/restaurant_reviews.parquet",  
    "ignoreOtherFileFormats": false,  
    "autoCorrectCorruptDates": true  
  },  
  "accessControlList": {  
    "users": [  
      {  
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",  
        "permissions": [  
          "SELECT",  
          "ALTER"  
        ]  
      }  
    ]  
  }  
}
```

```

    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "city",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "state",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "categories",
    "type": {
      "name": "LIST",
      "subSchema": [

```

```

        {
            "type": {
                "name": "VARCHAR"
            }
        }
    ]
}
},
{
    "name": "review_count",
    "type": {
        "name": "BIGINT"
    }
},
{
    "name": "stars",
    "type": {
        "name": "DOUBLE"
    }
},
{
    "name": "attributes",
    "type": {
        "name": "STRUCT",
        "subSchema": [
            {
                "name": "Parking",
                "type": {
                    "name": "STRUCT",
                    "subSchema": [
                        {
                            "name": "garage",
                            "type": {
                                "name": "BOOLEAN"
                            }
                        }
                    ],
                }
            },
            {
                "name": "street",
                "type": {
                    "name": "BOOLEAN"
                }
            }
        ],
        {
            "name": "lot",
            "type": {
                "name": "BOOLEAN"
            }
        }
    ],
    {
        "name": "valet",
        "type": {
            "name": "BOOLEAN"
        }
    }
}
}

```



```

    ]
  }
},
{
  "name": "Accepts Credit Cards",
  "type": {
    "name": "BOOLEAN"
  }
},
{
  "name": "Wheelchair Accessible",
  "type": {
    "name": "BOOLEAN"
  }
},
{
  "name": "Price Range",
  "type": {
    "name": "BIGINT"
  }
}
]
}
},
{
  "name": "date",
  "type": {
    "name": "VARCHAR"
  }
}
],
"approximateStatisticsAllowed": false
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found

Update a Table

Update the specified table in Dremio.

Method and URL

Parameters

id Path String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

entityType Body String

Type of the catalog object. For tables, the entityType is `dataset`.

id Body String (UUID)

Unique identifier of the table that you want to update.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

path Body Array of String

Path of the table that you want to update, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the table itself as the last item in the array. Get the path from the table's children object in the response to a [Folder](#) request.

Example: ["Samples", "samples.dremio.com", "Dremio University", "restaurant_reviews.parquet"]

tag Body String (UUID) Optional

Unique identifier of the version of the table that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the table. If you do not provide a tag, Dremio automatically updates the most recent version of the table.

Example: cb2905bb-39c0-497f-ae74-4c310d534f25

type Body String

Type of dataset. For tables, the type is `PHYSICAL_DATASET`.

Example:

[accelerationRefreshPolicy](#) Object

Attributes that define the acceleration refresh policy for the table.

[format](#) Body String

Parameters that describe the table's format.

[accessControlList](#) Body String Optional

Enterprise only. Object used to specify which users and roles should have access to the

table and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

```
Example: {"users": [{ "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"] }, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE\ GRANTS"] } ], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"]} ] }
```

Parameters of the `accelerationRefreshPolicy` Object

activePolicyType Body String

Policy to use for refreshing Reflections that are defined on the source. For this option to take effect, the neverRefresh parameter must be set to `false`.

The possible values are:

- `NEVER`: The Reflections are never refreshed.
- `PERIOD`: Default. The Reflections are refreshed at the end of every period that is defined by refreshPeriodMs.
- `SCHEDULE`: The Reflections are refreshed according to the schedule that is set by refreshSchedule.
- `REFRESH_ON_DATA_CHANGES`: Reflections automatically refresh for underlying tables that are in Iceberg format when new snapshots are created after an update. If the Reflection refresh job finds no changes, then no data is updated. Reflections that are automatically updated based on Iceberg source table changes also update according to the source-level policy as the minimum refresh frequency. Only available for tables in Iceberg format.

refreshPeriodMs Body Integer

Refresh period to use for the data in all Reflections for the table. In milliseconds. Optional if you set activePolicyType to `PERIOD`. The default setting is `3600000` milliseconds or one hour, which is also the minimum amount of time that is supported.

Example: 3600000

refreshSchedule Body String

A cron expression that sets the schedule, in UTC time, according to which the Reflections that are defined on the source should be refreshed. Optional if you set activePolicyType to `SCHEDULE`. The default refreshSchedule setting is to refresh every day at 8:00 a.m.

Field	Allowed Values	Allowed Special Characters
---	---	---
Second	0 N/A	
Minute	0-59 N/A	
Hour	0-23 N/A	
Day of month	N/A * ?	
Month	N/A * ?	
Days of week	1-7 or SUN-SAT , - * ?	

| Special Character | Description |

| --- | --- |

| * | Used to specify all values for a field. For `Day of month`, specifies every day of the month. For `Month`, specifies every month. For `Days of week`, specifies every day of the week. |

| ? | Equivalent to *.

| , | Used to specify two or more days in the `Days of week` field. For example, `MON,WED,FRI`.

| - | Used to specify ranges in the `Days of week` field. For example, `1-3` is equivalent to `Sunday, Monday, and Tuesday`.

Examples:

`0 0 0 * * ?` : Refreshes every day at midnight.

`0 45 15 * * 1,4,7` : Refreshes at 15:45 on Sunday, Wednesday, and Saturday.

`0 15 7 ? * 2-6` : Refreshes at 7:15 on Monday and Friday.

gracePeriodMs Body Integer

Maximum age to allow for Reflection data used to accelerate queries. In milliseconds.

Example: 10800000

method Body String

Method to use for refreshing the data in Reflections. For tables that are in the Apache Iceberg format; Parquet datasets in filesystems; or Parquet datasets, Avro datasets, or non-transactional ORC datasets in AWS Glue, the value is `AUTO`. In this case, the method used depends on this algorithm:

The initial refresh of a Reflection is always a full refresh.

If the Reflection is created from a view that uses nested group-bys, joins, unions, or window functions, then a full refresh is performed.

If the changes to the base table are only appends, then an incremental refresh based on table snapshots is performed.

If the changes to the base table include non-append operations, then a partition-based incremental refresh is attempted.

If the partitions of the base table and the partitions of the Reflection are not compatible, or if either the base table or the Reflection is not partitioned, then a full refresh is performed.

Read [Refreshing Reflections](#) for more information.

Enum: AUTO, FULL, INCREMENTAL

Example: FULL

refreshField Body String

For the `INCREMENTAL` refresh method, the field to refresh for the table. Used only if the method is `INCREMENTAL`. This parameter applies only to tables that are not in the Apache Iceberg format.

Example: business_id

neverExpire Body Boolean

If the Reflection should never expire, `true`. Otherwise, `false`.

Example: false

neverRefresh Body Boolean

If the Reflection should never refresh, `true`. Otherwise, `false`.

Example: false

Parameters of the `format` Object

type Body String

Type of data in the table.

Enum: Delta, Excel, Iceberg, JSON, Parquet, Text, Unknown, XLS

skipFirstLine Body Boolean Optional

If Dremio should skip the first line in the table, set to `true`. Otherwise, set to `false` (default). Optional for Excel and Text types.

Example: true

extractHeader Body Boolean Optional

If Dremio should extract the table's column names from the first line of the file, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Excel and Text types.

Example: true

hasMergedCells Body Boolean Optional

If Dremio should expand merged cells in the table, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Excel type.

Example: true

fieldDelimiter Body String Optional

Character to use to indicate separate fields in the table. May be `,` for a comma (default), `\t` for a tab, `|` for a pipe, or a custom character. Optional for tables created from files or folders of the Text type.

quote Body String Optional

Character to use for quotes in the table. May be `"` for a double quote (default), `'` for a single quote, or a custom character. Optional for tables created from files or folders of the Text type.

comment Body String Optional

Character to use to indicate comments for the table. May be `#` for a number sign (default) or a custom character. Optional for tables created from files or folders of the Text type.

escapeBody String Optional

Character to use to indicate an escape for the table. May be `\"` for a double quote (default), `` ` `` for a back quote, `\\` for a backward slash, or a custom character. Optional for tables created from files or folders of the Text type.

lineDelimiter Body String Optional

Character to use to indicate separate lines for the table. May be `\\r\\n` for a carriage return and a new line (default), `\\n` for a new line, or a custom character. Optional for tables created from files or folders of the Text type.

Example:

autoGenerateColumnNames Body Boolean Optional

If Dremio should use the existing column names for the table columns, set to `true` (default). Otherwise, set to `false`. Optional for tables created from files or folders of the Text type.

Example: true

trimHeader Body Boolean Optional

If Dremio should trim column names to a specific number of characters when updating the table, set to `true`. Otherwise, set to `false` (default). Optional for tables created from files or folders of the Text type.

Example: true

Parameters of the `accessControlList` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the table and the specific privileges each user should have.

Example:

```
[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE\\_GRANTS"]}]
```

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the table and the specific privileges each role should have.

Example:

```
[{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]
```

Parameters of Objects in the `users` and `roles` Arrays

id Body String

Enterprise only. Unique identifier of the user or role that should have access to the table.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String

Enterprise only. List of privileges the user or role should have on the table. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/dbale4fe-6351-44d2-a3e0-7aa20e782bf3' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "dbale4fe-6351-44d2-a3e0-7aa20e782bf3",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ],
  "type": "PHYSICAL_DATASET",
  "format": {
    "type": "Text",
    "fieldDelimiter": ",",
    "skipFirstLine": false,
    "extractHeader": true,
    "quote": "\"",
    "comment": "#",
    "escape": "\\",
    "lineDelimiter": "\r\n",
    "autoGenerateColumnNames": true,
    "trimHeader": true
  }
}'
```

Response

```
{
  "entityType": "dataset",
  "id": "dbale4fe-6351-44d2-a3e0-7aa20e782bf3",
  "type": "PHYSICAL_DATASET",
  "path": [
    "Samples",
    "samples.dremio.com",
    "Dremio University",
    "airbnb_listings.csv"
  ]
}
```

```

],
"createdAt": "2024-01-23T21:26:59.568Z",
"tag": "fc1707df-35a1-45c1-87d7-5f66fb11a729",
"format": {
  "type": "Text",
  "ctime": 0,
  "isFolder": false,
  "location": "/samples.dremio.com/Dremio University/airbnb_listings.csv",
  "fieldDelimiter": ",",
  "skipFirstLine": false,
  "extractHeader": true,
  "quote": "\"",
  "comment": "#",
  "escape": "\\",
  "lineDelimiter": "\r\n",
  "autoGenerateColumnNames": true,
  "trimHeader": true
},
"accessControlList": {},
"owner": {
  "ownerId": "c590ed7f-7142-4e1f-ba7d-94173afdc9a3",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "listing_url",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "scrape_id",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "last_scraped",
    "type": {
      "name": "VARCHAR"
    }
  },
  {
    "name": "name",
    "type": {
      "name": "VARCHAR"
    }
  }
],

```



```

    {
      "name": "summary",
      "type": {
        "name": "VARCHAR"
      }
    },
    {
      "name": "reviews_per_month",
      "type": {
        "name": "VARCHAR"
      }
    }
  ],
  "approximateStatisticsAllowed": false
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Refresh the Reflections on a Table

Refresh the Reflections associated with the specified table.

note

Refreshing a table's Reflections does not refresh its metadata. Read [Refreshing Metadata](#) to learn how to refresh table metadata. Read [Refreshing Reflections](#) for more information about refreshing Reflections.

Method and URL

```
POST /api/v3/catalog/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Revert a Table to a File or Folder

Revert a table in a source to change the data in the table back to its original format, file or folder. For more information, read [Formatting Data to a Table](#) and [Removing Formatting on Data](#)

note

If a table is saved in your home space, the revert request will delete the table entirely. The revert endpoint only changes a table back to a file or folder if the table is saved in a source.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the table that you want to revert to a file or folder.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

File](/current/reference/api/catalog/file)[Next

User-Defined Function](/current/reference/api/catalog/user-defined-function)

Table Attributes

Format a File or Folder as a Table

Parameters

Example

Response Status Codes

Retrieve a Table by ID

Parameters

Example

Response Status Codes

Retrieve a Table by Path

Parameters

Example

Response Status Codes

Update a Table

Parameters

Example

Response Status Codes

Refresh the Reflections on a Table

Parameters

Example

Response Status Codes

Revert a Table to a File or Folder

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/view/>

Version: current [26.x]

On this page

View

Use the Catalog API to retrieve, create, update, and delete views.

View Object

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
```

```

    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "permissions": [
    "READ",
    "WRITE",
    "ALTER_REFLECTION",
    "SELECT",
    "ALTER",
    "VIEW_REFLECTION",
    "MODIFY",
    "MANAGE_GRANTS",
    "CREATE_TABLE",
    "DROP",
    "EXTERNAL_QUERY",
    "INSERT",
    "TRUNCATE",
    "DELETE",
    "UPDATE",
    "EXECUTE",
    "CREATE_SOURCE",
    "ALL"
  ],
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  }
},

```

```

"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "other",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {

```

```

        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ]
}

```

View Attributes

entityType String

Type of the catalog object. For views, the entityType is `dataset`.

Example: dataset

id String (UUID)

Unique identifier of the view.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

type String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

Example: VIRTUAL_DATASET

path Array of String

Path of the view within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the view itself as the last item in the array.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

createdAt String

Date and time that the view was created, in UTC format.

Example: 2022-11-17T18:31:23.236Z

isMetadataExpired Boolean

If true, the metadata of the tables that the view is defined on needs to be refreshed. To

refresh it, run the ALTER VIEW command, using the clause REFRESH METADATA.

If false, the metadata can still be used for planning queries against the view.

If NULL, metadata has never yet been collected for the tables that the view is defined on.

lastMetadataRefreshAt String

Date and time that the metadata of the tables that the view is defined on was last refreshed. In UTC format.

Example: 2024-01-31T09:50:01.012Z

tag String (UUID)

Unique identifier of the version of the view. Dremio changes the tag whenever the view changes and uses the tag to ensure that PUT requests apply to the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql String

SQL query used to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Array of String

Context for the SQL query used to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Object

Enterprise only. Information about users and roles with access to the view and the specific privileges each user or role has. May include an array of users, an array of roles, or both, depending on the configured access and privileges. The accessControlList array is empty if view-specific access control privileges are not set.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT", "ALTER"] }, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"] }], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT", "ALTER"]}]}

permissions Array of String

Enterprise-only. List of the privileges that you have on the view. Only appears in the response if the request URL includes the `permissions` query parameter. For more information, read [Privileges](#).

Example:

["READ", "WRITE", "ALTER_REFLECTION", "SELECT", "ALTER", "VIEW_REFLECTION", "MODIFY", "MANAGE_GRANTS", "CREATE_TABLE", "DROP", "EXTERNAL_QUERY", "INSERT", "TRUNCATE", "DELETE", "UPDATE", "EXECUTE", "CREATE_SOURCE", "ALL"]

owner String

Information about the view's owner.

Example: `{"ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8","ownerType": "USER"}`

fields Array of Object

Attributes that represent the dataset schema.

Attributes of the `accessControlList` Object

users Array of Object

Enterprise only. List of users with access to the view and the specific privileges each user has.

Example: `[{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions": ["SELECT", "ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions": ["SELECT", "ALTER", "MANAGE_GRANTS"]}]`

roles Array of Object

Enterprise only. List of roles whose members have access to the view and the specific privileges each role has.

Example: `[{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions": ["SELECT","ALTER"]}]`

Attributes of Objects in the `users` and `roles` Arrays

id String

Enterprise only. Unique identifier of the user or role with access to the view.

Example: `c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3`

permissions Array of String

Enterprise only. List of privileges the user or role has on the view. For more information, read [Privileges](#).

Example: `["SELECT","ALTER"]`

Attributes of the `owner` Object

ownerId String (UUID)

Unique identifier of the view's owner.

Example: `30fca499-4abc-4469-7142-fc8dd29acac8`

ownerType String

Type of owner of the view.

Enum: USER, ROLE

Example: USER

Attributes of Objects in the `fields` Array

name String

Name of the view field.

Example: pickup_datetime

type Object

Information about the view field.

Attributes of the `type` Object

name String

Name of the view field's type.

Enum: STRUCT, LIST, UNION, INTEGER, BIGINT, FLOAT, DOUBLE, VARCHAR, VARBINARY, BOOLEAN, DECIMAL, TIME, DATE, TIMESTAMP, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

Example: TIMESTAMP

precision Integer

Total number of digits in the number. Included only for the `DECIMAL` type.

Example: 38

scale Integer

Number of digits to the right of the decimal point. Included only for the `DECIMAL` type.

Example: 2

subSchema Array of Object

List of objects that represent the field's composition. For example, a field composed of data about a restaurant might have a subSchema with an object for parking options, another for payment methods, and so on. subSchemas may be nested within other subSchemas. subSchema is listed only for the `STRUCT`, `LIST`, and `UNION` types.

Attributes of Objects in the `subSchema` Array

name String

Name for the subSchema object.

Example: cash

type Object

Object that contains a `name` attribute that provides the field's type.

Example: {"name": "BOOLEAN"}

Create a View

Create a view from a table in Dremio.

Method and URL

```
POST /api/v3/catalog
```

Parameters

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by a name for the view itself as the last item in the array. The name of the view cannot include the following special characters: `/`, `:`, `[`, or `]`. Views can only be created in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

sql Body String

SQL query to use to create the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC

sqlContext Body Array of String

Context for the SQL query to use to create the view.

Example: ["Samples", "samples.dremio.com"]

accessControlList Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Example: {"users": [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"]}], "roles": [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"]}]}

Parameters of the `accessControlList` Object

users [Body] Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3", "permissions": ["SELECT","ALTER"]}, {"id": "30fca499-4abc-4469-7142-fc8dd29acac8", "permissions": ["SELECT","ALTER","MANAGE_GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390", "permissions": ["SELECT","ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
```

```

"type": "VIRTUAL_DATASET",
"sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
"sqlContext": [
  "Samples",
  "samples.dremio.com"
],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
}
}'

```

Response

```

{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY

```

```

trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  },
  "fields": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "passenger_payment_method",
      "type": {
        "name": "STRUCT",

```

```

    "subSchema": [
      {
        "name": "cash",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "credit-debit",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "payment-app",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "other",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a View by ID

Retrieve a view by specifying the view's `id` value.

Method and URL

```
GET /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to retrieve.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <<dremioAccessToken>>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
```



```

"path": [
  "Business",
  "Transportation",
  "NYC-taxi-trips-short-distance"
],
"createdAt": "2022-11-17T18:31:23.236Z",
"isMetadataExpired": false,
"lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
"tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
"sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY
trip_distance_mi ASC",
"sqlContext": [
  "Samples",
  "samples.dremio.com"
],
"accessControlList": {
  "users": [
    {
      "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    },
    {
      "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
      "permissions": [
        "SELECT",
        "ALTER",
        "MANAGE_GRANTS"
      ]
    }
  ],
  "roles": [
    {
      "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
      "permissions": [
        "SELECT",
        "ALTER"
      ]
    }
  ]
},
"owner": {
  "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
  "ownerType": "USER"
},
"fields": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  }
],

```

```

{
  "name": "passenger_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "passenger_payment_method",
  "type": {
    "name": "STRUCT",
    "subSchema": [
      {
        "name": "cash",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "credit-debit",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "payment-app",
        "type": {
          "name": "BOOLEAN"
        }
      },
      {
        "name": "other",
        "type": {
          "name": "BOOLEAN"
        }
      }
    ]
  }
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
}

```

```
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
]
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a View by Path

Retrieve a view by specifying the view's path.

Method and URL

```
GET /api/v3/catalog/by-path/{path}
```

Parameters

path Path String

View's location within Dremio, using forward slashes as separators. For example, for the "NYC-taxi-trips" view in the "samples.dremio.com" folder within the space "Transportation," the path is `Transportation/samples.dremio.com/NYC-taxi-trips`. If the name of any component in the path includes special characters for URLs, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: Business/Transportation/NYC-taxi-trips-short-distance

include Query String Optional

Include a non-default attribute in the response. The available value for the include query parameter is `permissions`. For more information, read [include and exclude Query Parameters](#).

Example: ?include=permissions

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/by-path/Business/Transportation/NYC-taxi-trips-short-distance' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "createdAt": "2022-11-17T18:31:23.236Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT * FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
```

```

        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
            "SELECT",
            "ALTER"
        ]
    },
    ],
    },
    "owner": {
        "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "ownerType": "USER"
    },
    "fields": [
        {
            "name": "pickup_datetime",
            "type": {
                "name": "TIMESTAMP"
            }
        },
        {
            "name": "passenger_count",
            "type": {
                "name": "BIGINT"
            }
        },
        {
            "name": "passenger_payment_method",
            "type": {
                "name": "STRUCT",
                "subSchema": [
                    {
                        "name": "cash",
                        "type": {
                            "name": "BOOLEAN"
                        }
                    },
                    {
                        "name": "credit-debit",
                        "type": {
                            "name": "BOOLEAN"
                        }
                    },
                    {
                        "name": "payment-app",
                        "type": {
                            "name": "BOOLEAN"
                        }
                    },
                    {
                        "name": "other",
                        "type": {
                            "name": "BOOLEAN"
                        }
                    }
                ]
            }
        }
    ]
}

```

```

    ]
  }
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Update a View

Update a view in Dremio.

Method and URL

PUT /api/v3/catalog/{id}

Parameters

id Path String (UUID)

Unique identifier of the view that you want to update.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

entityType Body String

Type of the catalog object. For views, the entityType is `dataset`.

type Body String

Type of dataset. For views, type is `VIRTUAL_DATASET`.

path Body Array of String

Path of the location where you want to save the updated view within Dremio, expressed as an array. The path consists of the space, followed by any folder and subfolders, followed by the name for the view itself as the last item in the array. Views can only be saved in spaces.

Example: ["Business", "Transportation", "NYC-taxi-trips-short-distance"]

tag Body String (UUID) Optional

Unique identifier of the version of the view that you want to update. If you provide a tag in the request body, Dremio uses the tag to ensure that you are requesting to update the most recent version of the view. If you do not provide a tag, Dremio automatically updates the most recent version of the view.

Example: f90d1526-e64b-47b1-9ab0-d25df5247cab

sql Body String

SQL query to use to update the view.

Example: SELECT * FROM "NYC-taxi-trips" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC

sqlContext Body Array of String

Context for the SQL query to use for the updated view.

Example: ["Samples", "samples.dremio.com"]

accessControllist Body Object Optional

Enterprise only. Object used to specify which users and roles should have access to the view and the specific privileges each user or role should have. May include an array of users, an array of roles, or both.

Parameters of the `accessControllist` Object

users Body Array of Object Optional

Enterprise only. List of users who should have access to the view and the specific privileges each user should have.

Example: [{"id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3","permissions":["SELECT","ALTER"]},{ "id": "30fca499-4abc-4469-7142-fc8dd29acac8","permissions":["SELECT","ALTER","MANAGE\\GRANTS"]}]

roles Body Array of Object Optional

Enterprise only. List of roles whose members should have access to the view and the specific privileges each role should have.

Example: [{"id": "76a9884b-aea5-46d5-a73a-000edf23f390","permissions":["SELECT","ALTER"]}]

Parameters of Objects in the `users` and `roles` Arrays

id Body String Optional

Enterprise only. Unique identifier of the user or role who should have access to the view.

Example: c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3

permissions Body Array of String Optional

Enterprise only. List of privileges the user or role should have on the view. For more information, read [Privileges](#).

Example: ["SELECT", "ALTER"]

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-short-distance"
  ],
  "type": "VIRTUAL_DATASET",
  "tag": "f90d1526-e64b-47b1-9ab0-d25df5247cab",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ]
}'
```


Response

```
{
  "entityType": "dataset",
  "id": "ef99ab32-89ca-4d1c-9e91-2c8be861bb8a",
  "type": "VIRTUAL_DATASET",
  "path": [
    "Business",
    "Transportation",
    "NYC-taxi-trips-by-distance"
  ],
  "createdAt": "2023-01-20T15:26:39.780Z",
  "isMetadataExpired": false,
  "lastMetadataRefreshAt": "2024-01-31T09:50:01.012Z",
  "tag": "7cab1a42-8835-4d31-827b-fedee1ad38d1",
  "sql": "SELECT trip_distance_mi, fare_amount, tip_amount FROM \"NYC-taxi-trips\" WHERE trip_distance_mi <= 2.0 ORDER BY trip_distance_mi DESC",
  "sqlContext": [
    "Samples",
    "samples.dremio.com"
  ],
  "accessControlList": {
    "users": [
      {
        "id": "c590ed7f-b2b4-4e1f-ba7d-94173afdc9a3",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      },
      {
        "id": "30fca499-4abc-4469-7142-fc8dd29acac8",
        "permissions": [
          "SELECT",
          "ALTER",
          "MANAGE_GRANTS"
        ]
      }
    ],
    "roles": [
      {
        "id": "76a9884b-aea5-46d5-a73a-000edf23f390",
        "permissions": [
          "SELECT",
          "ALTER"
        ]
      }
    ]
  },
  "owner": {
    "ownerId": "30fca499-4abc-4469-7142-fc8dd29acac8",
    "ownerType": "USER"
  },
  "fields": [
    {
```

```

    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "passenger_payment_method",
    "type": {
      "name": "STRUCT",
      "subSchema": [
        {
          "name": "cash",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "credit-debit",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "payment-app",
          "type": {
            "name": "BOOLEAN"
          }
        },
        {
          "name": "other",
          "type": {
            "name": "BOOLEAN"
          }
        }
      ]
    }
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  }

```

```

    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ]
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Refresh the Reflections on a View

Refresh the Reflections associated with the specified view.

Read [Refreshing Reflections](#) to learn how refreshing works.

Method and URL

```
POST /api/v3/catalog/{id}/refresh
```

Parameters

id Path String (UUID)

Unique identifier for the view you want to refresh.

Example: c9c11d32-0576-4200-5a5b-8c7229cb3d72

Example

Request

```
curl -X POST 'https://api.dremio.cloud//api/v3/catalog/c9c11d32-0576-4200-5a5b-8c7229cb3d72/refresh' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Delete a View

Delete the specified view.

Method and URL

```
DELETE /api/v3/catalog/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the view that you want to delete.

Example: ef99ab32-89ca-4d1c-9e91-2c8be861bb8a

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/catalog/ef99ab32-89ca-4d1c-9e91-2c8be861bb8a' \
-H 'Authorization: Bearer <dremioAccessToken>' \
-H 'Content-Type: application/json'
```

Response

No response

Response Status Codes

204 No Content

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

User-Defined Function](/current/reference/api/catalog/user-defined-function)[Next

Lineage](/current/reference/api/catalog/lineage)

View Attributes

Create a View

Parameters

Example

Response Status Codes

Retrieve a View by ID

Parameters

Example

Response Status Codes

Retrieve a View by Path

Parameters

Example

Response Status Codes

Update a View

Parameters

Example

Response Status Codes

Refresh the Reflections on a View

Parameters

Example

Response Status Codes

Delete a View

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/lineage/>

Version: current [26.x]

On this page

Lineage Enterprise

Use the Catalog API to retrieve lineage information about datasets (tables and views). The lineage object includes information about the dataset's sources, parent objects, and child objects.

Lineage Object

```
{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
      "createdAt": "2022-02-14T21:57:48.794Z"
    }
  ],
  "parents": [
    {
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zips.json"
      ],
      "tag": "MAntohVzwLw=",

```

```

    "type": "DATASET",
    "datasetType": "PROMOTED",
    "createdAt": "2023-01-18T18:49:09.669Z"
  },
  "children": [
    {
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
      "path": [
        "@dremio",
        "NYC_zip"
      ],
      "tag": "0WKrfpEKzW4=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T02:11:46.344Z"
    },
    {
      "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
      "path": [
        "@dremio",
        "Chicago_zip"
      ],
      "tag": "gsaDW5h4GCs=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T00:09:12.461Z"
    }
  ]
}

```

Lineage Attributes

sources Array of Object

Information about the sources the dataset uses. Each object in the sources array represents one source.

Example: [{ "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2", "path": ["Samples"], "tag": "lz1v71CeTQY=", "type": "CONTAINER", "containerType": "SOURCE", "createdAt": "2023-02-14T21:57:48.794Z" }]

parents Array of Object

Information about the parent objects for the dataset. Each object in the parents array represents one parent object. If a view represents a join of two other datasets, the parents array includes the two joined datasets. The parents array is empty if the dataset does not have parent objects.

Example: [{ "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752", "path": ["Samples", "samples.dremio.com", "zips.json"], "tag": "MAntohVzwLw=", "type": "DATASET", "datasetType": "PROMOTED", "createdAt": "2023-01-18T18:49:09.669Z" }]

children Array of Object

Information about other catalog objects that reference the dataset. Each object in the children array represents one child object. The children array is empty if the dataset does not have child objects.

Example:

```
[{"id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75", "path": ["@dremio", "NYC\\_zip"], "tag": "OWKrfpEKzW4=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-25T02:11:46.344Z"}, {"id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0", "path": ["@dremio", "Chicago\\_zip"], "tag": "gsaDW5h4GCs=", "type": "DATASET", "datasetType": "VIRTUAL", "createdAt": "2023-01-25T00:09:12.461Z"}]
```

Attributes of Objects in the `sources` Array

id String (UUID)

Unique identifier of the source associated with the dataset.

Example: 21077e5d-fe6f-4a29-843f-58fa3acb17c2

path Array of String

Path of the source within Dremio, expressed as an array.

Example: ["Samples"]

tag String

Unique identifier of the version of the source. Dremio changes the tag whenever the source changes.

Example: lz1v71CeTQY=

type String

Type of source. For sources in lineage responses, the type is `CONTAINER`.

Example: CONTAINER

containerType String

Type of container for the source.

Enum: HOME, SOURCE

Example: SOURCE

createdAt String

Date and time that the source was created, in UTC format. Not included for sources with the containerType `HOME`.

Example: 2022-02-14T21:57:48.794Z

Attributes of Objects in the `parents` Array

id String (UUID)

Unique identifier of the parent object.

Example: 3419fa3a-b5b3-4438-b864-a27ec4e18752

path Array of String

Path of the parent object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the parent object itself as the last item in the array.

Example: ["Samples","samples.dremio.com","zips.json"]

tag String

Unique identifier of the version of the parent object. Dremio changes the tag whenever the parent object changes.

Example: MAntohVzwLw=

type String

Type of parent object. For parent objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the parent object. If the parent object is a table, `PROMOTED`. If the parent object is a view, `VIRTUAL`.

Enum: PROMOTED, VIRTUAL

Example: PROMOTED

createdAt String

Date and time that the parent object was created, in UTC format.

Example: 2023-01-18T18:49:09.669Z

Attributes of Objects in the `children` Array

id String (UUID)

Unique identifier of the child object.

Example: 170e211e-4235-4d8d-acb5-3d4dbfe99c75

path Array of String

Path of the child object within Dremio, expressed as an array. The path consists of the source or space, followed by any folder and subfolders, followed by the child object itself as the last item in the array.

Example: ["@dremio","NYC_zip"]

tag String

Unique identifier of the version of the child object. Dremio changes the tag whenever the child object changes.

Example: OWKrfpEKzW4=

type String

Type of child object. For child objects in lineage responses, the type is `DATASET`.

Example: DATASET

datasetType String

Dataset type for the child object. For child objects in lineage responses, the datasetType is `VIRTUAL`.

Example: VIRTUAL

createdAt String

Date and time that the child object was created, in UTC format.

Example: 2023-01-25T02:11:46.344Z

Retrieve Lineage Information About a Dataset

Retrieve lineage information about the specified dataset.

Method and URL

```
GET /api/v3/catalog/{id}/graph
```

Parameters

id Path String (UUID)

Unique identifier of the dataset whose lineage you want to retrieve.

Example: d69b25a3-31c8-4d55-a7cc-dfee2290779b

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/d69b25a3-31c8-4d55-a7cc-dfee2290779b/graph' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```

{
  "sources": [
    {
      "id": "21077e5d-fe6f-4a29-843f-58fa3acb17c2",
      "path": [
        "Samples"
      ],
      "tag": "Iz1v71CeTQY=",
      "type": "CONTAINER",
      "containerType": "SOURCE",
      "createdAt": "2022-02-14T21:57:48.794Z"
    }
  ],
  "parents": [
    {
      "id": "3419fa3a-b5b3-4438-b864-a27ec4e18752",
      "path": [
        "Samples",
        "samples.dremio.com",
        "zips.json"
      ],
      "tag": "MAntohVzwLw=",
      "type": "DATASET",
      "datasetType": "PROMOTED",
      "createdAt": "2023-01-18T18:49:09.669Z"
    }
  ],
  "children": [
    {
      "id": "170e211e-4235-4d8d-acb5-3d4dbfe99c75",
      "path": [
        "@dremio",
        "NYC_zip"
      ],
      "tag": "0WKrfpEKzW4=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T02:11:46.344Z"
    },
    {
      "id": "7f79c068-a3c3-4af7-8cd4-35896ef0a0e0",
      "path": [
        "@dremio",
        "Chicago_zip"
      ],
      "tag": "gsaDW5h4GCs=",
      "type": "DATASET",
      "datasetType": "VIRTUAL",
      "createdAt": "2023-01-25T00:09:12.461Z"
    }
  ]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

View](/current/reference/api/catalog/view)[Next

Tag](/current/reference/api/catalog/tag)

Lineage Attributes

Retrieve Lineage Information About a Dataset

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/tag/>

Version: current [26.x]

On this page

Tag

Use the Catalog API to create, update, and retrieve tags.

Tag Object

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VJ3ijXH4m6k="
}
```

Tag Attributes

tags Array of String

List of tags that apply to the dataset.

Example: ["NYC","taxi","2023"]

version String

Unique identifier of the set of tags. Dremio changes the version whenever any of the tags change and uses the version value to ensure that updates apply to the most recent version of the set of tags.

Example: VJ3ijXH4m6k=

Create Tags

Create one or more tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset for which you want to add tags.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. Tags are case-insensitive. Each tag can be listed only once for each dataset. Each tag can have a maximum of 128 characters. Tags cannot include the following special characters: `/`, `:`, `[`, or ` `.

Example: ["NYC","taxi","2023"]

Example

Request

```
curl -X POST \
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023"]
}'
```

Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve Tags

Retrieve the tags applied to the specified dataset.

Method and URL

```
GET /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example

Request

```
curl -X GET
'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023"
  ],
  "version": "VM3ijXH4m6k="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Update Tags

Update the tags for the specified dataset.

Method and URL

```
POST /api/v3/catalog/{dataset-id}/collaboration/tag
```

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

List of tags to apply to the dataset. If you want to keep any of the existing tags, include them in the tags array. Tags are case-insensitive and must be distinct (in other words, list each tag only once for each dataset). Each tag may have a maximum of 128 characters. Tags cannot include the following special characters: `/`, `:`, `[`, or ` `.

Example: ["NYC","taxi","2023","archived"]

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are updating the most recent version of the tags.

Example: VM3ijXH4m6k=

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": ["NYC", "taxi", "2023", "archived"],
  "version": "VM3ijXH4m6k="
}'
```

Response

```
{
  "tags": [
    "NYC",
    "taxi",
    "2023",
    "archived"
  ],
  "version": "yiZSE++9wiU="
}
```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict

Delete Tags

Delete the tags for the specified dataset.

note

Deleting tags means sending an empty array to replace the existing tags with no tags. The tag object will still exist, but it will contain an empty `tags` array and no tags will appear for the dataset in the Dremio UI.

Method and URL

POST /api/v3/catalog/{dataset-id}/collaboration/tag

Parameters

dataset-id Path String (UUID)

Unique identifier of the dataset whose tags you want to remove.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

tags Body Array of String

Empty array to represent deletion of all tags for the dataset.

Example: []

version Body String

Unique identifier of the most recent set of tags. Dremio uses the version value to ensure that you are deleting tags from the most recent version.

Example: yiZSE++9wiU=

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/tag' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tags": [],
  "version": "yiZSE++9wiU="
}'
```

Response

```
{
  "tags": [],
  "version": "wuTAKuRcVas="
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[Previous

Lineage](/current/reference/api/catalog/lineage)[Next

Wiki](/current/reference/api/catalog/wiki)

Tag Attributes

Create Tags

Parameters

Example

Response Status Codes

Retrieve Tags

Parameters

Example

Response Status Codes

Update Tags

Parameters

Example

Response Status Codes

Delete Tags

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/wiki/>

Version: current [26.x]

On this page

Wiki

Use the Catalog API to create, update, and retrieve the wiki for a source, space, or dataset.

Wiki Object

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.\nHere is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",
  "version": 4
}
```

Wiki Attributes

text String

Text displayed in the wiki, formatted with GitHub-flavored Markdown.

version Integer

Number for the most recent version of the wiki, starting with `0`. Dremio increments the value by 1 each time the wiki changes and uses the version value to ensure that updates apply to the most recent version of the wiki.

Example: 4

Create a Wiki

Create a wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset for which you want to add the wiki.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki. Use GitHub-flavored Markdown for wiki formatting and ``\n`` for new lines and blank lines. Each wiki may have a maximum of 100,000 characters.

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum."
}'
```

Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted
list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n###
Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second
item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that
includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve a Wiki

Retrieve the wiki for the specified source, space, or dataset.

Method and URL

```
GET /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to retrieve.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "text": "# Testspace Wiki\nThis is an example wiki for a catalog object in Dremio. Here is some text in bold. Here is some text in italics.\n\nHere is an example excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.\n\n\n## Heading Level 2\n\nHere is a bulleted list:\n* An item in a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted list\n\n\n### Heading Level 3\n\nHere is a numbered list:\n1. An item in a numbered list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere is a sentence that includes an [external link to https://dremio.com](https://dremio.com).\n\nHere is an image:\n\n\n\nUt
```

```
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Update a Wiki

Update the wiki for the specified source, space, or dataset.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to update.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Text to display in the wiki, formatted with GitHub-flavored Markdown.

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are updating the most recent version of the wiki.

Example: 0

Example

Request

```
curl -X POST
```

```
'https://{hostname}/api/v3/catalog/lbcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\n\nHere is an update to the bulleted list:\n* An item in
a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted
list\n\n\n### Heading Level 3\n\n\nHere is a numbered list:\n1. An item in a numbered
list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere
is a sentence that includes an [external link to
https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 0
}'
```

Response

```
{
  "text": "# New Title Wiki\nThis is an example wiki for a catalog object in Dremio.
Here is some text in bold. Here is some text in italics.\n\nHere is an example
excerpt with quotation formatting:\n\n> Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua.\n\n\n## Heading Level 2\n\n\nHere is an update to the bulleted list:\n* An item in
a bulleted list\n* A second item in a bulleted list\n* A third item in a bulleted
list\n\n\n### Heading Level 3\n\n\nHere is a numbered list:\n1. An item in a numbered
list\n1. A second item in a numbered list\n1. A third item in a numbered list\n\n\nHere
is a sentence that includes an [external link to
https://dremio.com](https://dremio.com).\n\nHere is an
image:\n\n\n\nUt
enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.",
  "version": 1
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Delete a Wiki

Delete the wiki for the specified source, space, or dataset.

note

Deleting the wiki entails sending an empty string to replace the existing wiki with no wiki. The wiki object will still exist, but it will contain an empty `text` value and no wiki will appear for the source, space, or dataset in the Dremio UI.

Method and URL

```
POST /api/v3/catalog/{object-id}/collaboration/wiki
```

Parameters

object-id Path String (UUID)

Unique identifier of the source, space, or dataset whose wiki you want to delete.

Example: 1bcab7b3-ee82-44c1-abcc-e86d56078d4d

text Body String

Empty string to represent deletion of the wiki.

Example: ""

version Body Integer

Number listed as the version value for the most recent existing wiki. Dremio uses the version value to ensure that you are deleting the most recent version of the wiki.

Example: 1

Example

Request

```
curl -X POST \
  'https://{hostname}/api/v3/catalog/1bcab7b3-ee82-44c1-abcc-e86d56078d4d/collaboration/wiki' \
  --header 'Authorization: Bearer <dremioAccessToken>' \
  --header 'Content-Type: application/json' \
  --data-raw '{
    "text": "",
    "version": 1
  }'
```


Response

```
{  
  "text": "",  
  "version": 2  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Was this page helpful?

[\[Previous](#)

[Tag\]/\(current/reference/api/catalog/tag\)\[Next](#)

[Grants\]/\(current/reference/api/catalog/grants\)](#)

[Wiki Attributes](#)

[Create a Wiki](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Retrieve a Wiki](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Update a Wiki](#)

[Parameters](#)

[Example](#)

[Response Status Codes](#)

[Delete a Wiki](#)

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/catalog/grants/>

Version: current [26.x]

On this page

Grants Enterprise

Use the Catalog API to grant user and role privileges on specific catalog objects.

Grants Object

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
      "firstName": "Jean",
      "lastName": "Smith",
      "email": "jean_smith@example.com"
    },
    {
      "privileges": [
        "ALTER",
        "SELECT"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "name": "examplerole"
    }
  ]
}
```

```
}  
]  
}
```

Grants Attributes

id String

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

availablePrivileges Array of String

List of available privileges on the catalog object.

Example:

```
["ALTER","DELETE","INSERT","MANAGE\ GRANTS","SELECT","TRUNCATE","UPDATE"]
```

grants Array of Object

Information about the privileges and grantees for the catalog object. If the grants array is empty, there are no explicit grants for the object.

note

An empty grants array does not mean no users have access to the object at all. For example, admin users implicitly have all privileges on all catalog objects, owners implicitly have all privileges on everything they own, and children objects inherit the grants for their parent objects.

Example: [{"privileges": ["ALTER","SELECT","MANAGE\ GRANTS"],"granteeType": "USER","id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0","name": "jeansmith","firstName": "Jean","lastName": "Smith","email": "jean\ smith@example.com"}, {"privileges": ["ALTER","SELECT"],"granteeType": "ROLE","id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889","name": "examplerole"}]

Attributes of Objects in the `grants` Array

privileges String

List of privileges granted to the user or role. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE\ GRANTS"]

granteeType String

Type of grantee.

Enum: USER, ROLE

Example: USER

id String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

name String

Name of the user or role.

Example: jeansmith

firstName String

For users, the user's first name. Not included for roles.

Example: Jean

lastName String

For users, the user's last name. Not included for roles.

Example: Smith

email String

For users, the user's email address. Not included for roles.

Example: jean_smith@example.com

Create or Updating Privilege Grants on a Catalog Object

Create or update the privileges granted to users and roles on the specified catalog object.

note

You must have the MANAGE GRANTS privilege to create or update privilege grants on catalog objects.

Method and URL

```
PUT /api/v3/catalog/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the Dremio catalog object.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

grants Body Array of Object

Array of objects that specify which users and roles should have privileges on the catalog object, as well as each user's and role's specific privileges. May include objects

for users, roles, or both.

Example: [{"privileges": ["ALTER","SELECT","MANAGE\ Grants"], "granteeType": "USER", "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"}, {"privileges": ["SELECT","ALTER"], "granteeType": "ROLE", "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"}]

Parameters of Objects in the `grants` Array

privileges Body Array of String

List of privileges to grant to the user or role. Use the [Privileges](#) endpoint to retrieve a list of available privileges on the catalog object type. For more information, read [Privileges](#).

Example: ["ALTER","SELECT","MANAGE\ Grants"]

granteeType Body String

Type of grantee.

Enum: USER, ROLE

Example: USER

id Body String

Unique identifier of the user or role.

Example: 27937a63-e7e5-4478-8d3c-4ad3f20d43c0

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0"
    },
    {
      "privileges": [
        "SELECT",
        "ALTER"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"
    }
  ]
}
```

```
"granteeType": "ROLE",  
  "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889"  
}  
]  
'
```

Response

No response

Response Status Codes

204 No Content

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve Privileges and Grantees on a Catalog Object

Retrieve information about the privileges granted to users and roles on the specified catalog object.

note

Use this endpoint in place of the Catalog API [Privileges](#) endpoint, which is deprecated. We expect to remove the Privileges endpoint by July 2025.

You must have the [MANAGE GRANTS](#) privilege to retrieve privilege grants on catalog objects.

Method and URL

```
GET /api/v3/catalog/{id}/grants
```

Parameters

id Path String (UUID)

Unique identifier of the object whose privilege grants you want to retrieve.

Example: 7f1c4660-cd7b-40d0-97d1-b8a6f431cbda

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/catalog/7f1c4660-cd7b-40d0-97d1-b8a6f431cbda/grants' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "7f1c4660-cd7b-40d0-97d1-b8a6f431cbda",
  "availablePrivileges": [
    "ALTER",
    "DELETE",
    "INSERT",
    "MANAGE_GRANTS",
    "SELECT",
    "TRUNCATE",
    "UPDATE"
  ],
  "grants": [
    {
      "privileges": [
        "ALTER",
        "SELECT",
        "MANAGE_GRANTS"
      ],
      "granteeType": "USER",
      "id": "27937a63-e7e5-4478-8d3c-4ad3f20d43c0",
      "name": "jeansmith",
      "firstName": "Jean",
      "lastName": "Smith",
      "email": "jean_smith@example.com"
    },
    {
      "privileges": [
        "ALTER",
        "SELECT"
      ],
      "granteeType": "ROLE",
      "id": "0f2d94e0-bb5e-4c03-8c6f-62d379d10889",
      "name": "examplerole"
    }
  ]
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Wiki]](/current/reference/api/catalog/wiki)

Grants Attributes

Create or Updating Privilege Grants on a Catalog Object

Parameters

Example

Response Status Codes

Retrieve Privileges and Grantees on a Catalog Object

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/engine-management/engine-configuration-options>

Version: current [26.x]

On this page

Engine Configuration Options Enterprise

Use the Engine Management API to manage your engines in Dremio:

****Engine Management**** - Covers the objects and operations for managing engines.

Engine Configuration - Covers the objects and operations for engine configuration options.

Engine Configuration Options

```
{
  "targetCpuCapacities": {
    "capacities": [
      {
        "name": "16C",
        "cpu": "16"
      },
      {
```



```

        "name": "32C",
        "cpu": "32"
    },
    "defaultCapacity": "32C"
},
"sizes": [
    {
        "name": "2XSmall",
        "pods": 1,
        "memory": "64Gi",
        "cpuScaleFactor": 0.5
    },
    {
        "name": "XSmall",
        "pods": 1,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "Small",
        "pods": 2,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "Medium",
        "pods": 4,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "Large",
        "pods": 8,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "XLarge",
        "pods": 16,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "2XLarge",
        "pods": 32,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    }
],
"resourceAllocationOffsets": {
    "offsets": [
        {
            "name": "reserve-0-0",

```

```

    "cpu": "0",
    "memory": "0Gi",
    "action": "Reserve"
  },
  {
    "name": "reserve-2-4",
    "cpu": "2",
    "memory": "4Gi",
    "action": "Reserve"
  },
  {
    "name": "reserve-2-8",
    "cpu": "2",
    "memory": "8Gi",
    "action": "Reserve"
  },
  {
    "name": "reserve-2-16",
    "cpu": "2",
    "memory": "16Gi",
    "action": "Reserve"
  }
],
"defaultOffset": "reserve-2-8"
},
"storage": {
  "spillStorageSizes": [
    {
      "name": "100GB",
      "storage": "100Gi"
    },
    {
      "name": "250GB",
      "storage": "250Gi"
    },
    {
      "name": "500GB",
      "storage": "500Gi"
    }
  ],
  "defaultSpillStorageSize": "100GB",
  "c3StorageSizes": [
    {
      "name": "100GB",
      "storage": "100Gi"
    },
    {
      "name": "250GB",
      "storage": "250Gi"
    },
    {
      "name": "500GB",
      "storage": "500Gi"
    }
  ]
}

```

```

    ],
    "defaultC3StorageSize": "100GB"
  },
  "idleTimeouts": {
    "durations": [
      "PT2H",
      "PT1H30M",
      "PT1H",
      "PT30M",
      "PT15M",
      "PT10M",
      "PT5M"
    ],
    "defaultDuration": "PT2H",
    "maximumDuration": "PT12H"
  }
}

```

Engine Configuration Options Attributes

targetCpuCapacities Object

Information about the possible CPU capacities options to configure engines and which is the default one.

Example: { "capacities": [{ "name": "16C", "cpu": "16" }, { "name": "32C", "cpu": "32" }], "defaultCapacity": "32C" }

sizes Array of Objects

Information about the possible sizes to configure engines.

Example: [{ "name": "2XSmall", "pods": 1, "memory": "64Gi", "cpuScaleFactor": 0.5 }, { "name": "XSmall", "pods": 1, "memory": "128Gi", "cpuScaleFactor": 1.0 }, { "name": "Small", "pods": 2, "memory": "128Gi", "cpuScaleFactor": 1.0 }, { "name": "Medium", "pods": 4, "memory": "128Gi", "cpuScaleFactor": 1.0 }, { "name": "Large", "pods": 8, "memory": "128Gi", "cpuScaleFactor": 1.0 }, { "name": "XLarge", "pods": 16, "memory": "128Gi", "cpuScaleFactor": 1.0 }, { "name": "2XLarge", "pods": 32, "memory": "128Gi", "cpuScaleFactor": 1.0 }]

resourceAllocationOffsets Object

Information about the possible pod resource allocation options to configure engines and which is the default one.

Example: { "offsets": [{ "name": "reserve-0-0", "cpu": "0", "memory": "0Gi", "action": "Reserve" }, { "name": "reserve-2-4", "cpu": "2", "memory": "4Gi", "action": "Reserve" }, { "name": "reserve-2-8", "cpu": "2", "memory": "8Gi", "action": "Reserve" }, { "name": "reserve-2-16", "cpu": "2", "memory": "16Gi", "action": "Reserve" }], "defaultOffset": "reserve-2-8" }

storage Object

Information all pod storage options.

Example: { "spillStorageSizes": [{ "name": "100GB", "storage": "100Gi" }, { "name": "250GB", "storage": "250Gi" }, { "name": "500GB", "storage": "500Gi" }], "defaultSpillStorageSize": "100GB", "c3StorageSizes": [{ "name": "100GB", "storage": "100Gi" }, { "name": "250GB", "storage": "250Gi" }, { "name": "500GB", "storage": "500Gi" }], "defaultC3StorageSize": "100GB" }

idleTimeouts Object

Information all idle timeout options, which is the default one, and the maximum duration.

Example: { "durations": ["PT2H", "PT1H30M", "PT1H", "PT30M", "PT15M", "PT10M", "PT5M"], "defaultDuration": "PT2H", "maximumDuration": "PT12H" }

Attributes of the `targetCpuCapacities` Object

capacities Array of Objects

The possible CPU capacities to configure engines.

Example: [{ "name": "16C", "cpu": "16" }, { "name": "32C", "cpu": "32" }]

defaultCapacity String

If not defined for the engine, this the default capacity for it.

Example: 32C

Attributes of Objects in the `capacities` Array

name String

The name of the cpu capacity.

Example: 32C

cpu String

The unadjusted cpu allocated to each executor pod of each replica of an engine. Expressed as a Kubernetes Quantity.

Example: 32

Attributes of Objects in the `sizes` Array

name String

The name of the engine size, to be used as the key.

Example: 2XSmall

Pods Integer

The number of executor pods created for each replica of an engine of this size.

Example: 8

memory String

The memory allocated to each executor pod of each replica of an engine of this size. Expressed as a Kubernetes Quantity.

Example: 64Gi

cpuScaleFactor Float

The scaling applied to the chosen TargetCpuCapacity for this engine size. Used to declare sizes that have smaller CPU allocations than those provided by Kubernetes nodes.

The value must be between 0.0 and 1.0. Defaults to 1.0.

Example: 1.0

Attributes of the `resourceAllocationOffsets` Object

offsets Array of Objects

The possible resource allocation offset options to configure engines.

Example: { "name": "reserve-0-0", "cpu": "0", "memory": "0Gi", "action": "Reserve" }, { "name": "reserve-2-4", "cpu": "2", "memory": "4Gi", "action": "Reserve" }, { "name": "reserve-2-8", "cpu": "2", "memory": "8Gi", "action": "Reserve" }, { "name": "reserve-2-16", "cpu": "2", "memory": "16Gi", "action": "Reserve" }

defaultOffset String

The default offset used for engines where a specific value is not specified.

Example: reserve-2-8

Attributes of the Objects in the `offsets` Array

name String

The name of the resource allocation offset, to be used as key.

Example: reserve-2-8

cpu String

The CPU adjustment to be made to each executor pod of each replica of an engine with this capacity.

Expressed as a Kubernetes Quantity.

Example: 0

memory String

The memory adjustment to be made to each executor pod of each replica of an engine with this capacity.

Expressed as a Kubernetes Quantity.

Example: 8Gi

action String

The offset type, where 'Reserve' subtracts the offset and 'OverCommit' adds it to the pod allocations.

Example: Reserve

Attributes of the `storage` Object

spillStorageSizes Array of Objects

The set of all spill storage size options.

Example: [{ "name": "100GB", "storage": "100Gi" }, { "name": "250GB", "storage": "250Gi" }, { "name": "500GB", "storage": "500Gi" }]

defaultSpillStorageSize String

The default spill storage size to used for engines where a specific value is not specified.

Example: 100GB

c3StorageSizes Array of Objects

The set of all C3 storage size options.

Example: [{ "name": "100GB", "storage": "100Gi" }, { "name": "250GB", "storage": "250Gi" }, { "name": "500GB", "storage": "500Gi" }]

defaultC3StorageSize String

The default C3 storage size to used for engines where a specific value is not specified.

Example: 100GB

Attributes of the Objects in the `spillStorageSizes` Array

name String

The name of the storage size, to be used as the key.

Example: 100GB

storage String

The storage allocated to each executor pod of each replica of an engine. Expressed as a Kubernetes Quantity.

Example: 128Gi

Attributes of the Objects in the `c3StorageSizes` Array

name String

The name of the storage size, to be used as the key.

Example: 100GB

storage String

The storage allocated to each executor pod of each replica of an engine.
Expressed as a Kubernetes Quantity.

Example: 128Gi

Attributes of the `idleTimeouts` Object

durations Array of String

The set of all idle timeout options.

Example: ["PT2H", "PT1H30M", "PT1H", "PT30M", "PT15M", "PT10M", "PT5M"]

defaultDuration String

The default idle timeout to used for engines where a specific value is not specified.

Example: PT1H

maximumDuration String

The maximum idle timeout that can be specified by the user.

Example: PT12H

Retrieve the Configuration Options

Retrieve all engine configuration options.

Method and URL

```
GET /api/v3/engines/configuration/options
```

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/engines/configuration/options' \
--header 'Authorization: Bearer <PersonalAccessToken>
```

Response

```
{
  "targetCpuCapacities": {
    "capacities": [
      {
        "name": "16C",
```

```

        "cpu": "16"
    },
    {
        "name": "32C",
        "cpu": "32"
    }
],
"defaultCapacity": "32C"
},
"sizes": [
    {
        "name": "2XSmall",
        "pods": 1,
        "memory": "64Gi",
        "cpuScaleFactor": 0.5
    },
    {
        "name": "XSmall",
        "pods": 1,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "Small",
        "pods": 2,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "Medium",
        "pods": 4,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "Large",
        "pods": 8,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "XLarge",
        "pods": 16,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    },
    {
        "name": "2XLarge",
        "pods": 32,
        "memory": "128Gi",
        "cpuScaleFactor": 1.0
    }
],
"resourceAllocationOffsets": {

```



```

"offsets": [
  {
    "name": "reserve-0-0",
    "cpu": "0",
    "memory": "0Gi",
    "action": "Reserve"
  },
  {
    "name": "reserve-2-4",
    "cpu": "2",
    "memory": "4Gi",
    "action": "Reserve"
  },
  {
    "name": "reserve-2-8",
    "cpu": "2",
    "memory": "8Gi",
    "action": "Reserve"
  },
  {
    "name": "reserve-2-16",
    "cpu": "2",
    "memory": "16Gi",
    "action": "Reserve"
  }
],
"defaultOffset": "reserve-2-8"
},
"storage": {
  "spillStorageSizes": [
    {
      "name": "100GB",
      "storage": "100Gi"
    },
    {
      "name": "250GB",
      "storage": "250Gi"
    },
    {
      "name": "500GB",
      "storage": "500Gi"
    }
  ],
  "defaultSpillStorageSize": "100GB",
  "c3StorageSizes": [
    {
      "name": "100GB",
      "storage": "100Gi"
    },
    {
      "name": "250GB",
      "storage": "250Gi"
    },
    {

```

```

        "name": "500GB",
        "storage": "500Gi"
    },
    ],
    "defaultC3StorageSize": "100GB"
},
"idleTimeouts": {
    "durations": [
        "PT2H",
        "PT1H30M",
        "PT1H",
        "PT30M",
        "PT15M",
        "PT10M",
        "PT5M"
    ],
    "defaultDuration": "PT2H",
    "maximumDuration": "PT12H"
}
}

```

Response Status Codes

201 Default configuration

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Retrieve the Deployment Type

Retrieve the deployment type.

Method and URL

```
GET /api/v3/engines/configuration/deployment-type
```

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/engines/configuration/deployment-type' \
--header 'Authorization: Bearer <PersonalAccessToken>'
```

Response

```
{  
  "type": "KUBERNETES"  
}
```

Response Status Codes

200 Supported cluster type

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Engine Management](/current/reference/api/engine-management/)[Next

External Token Providers](/current/reference/api/external-token-providers/)

Engine Configuration Options Attributes

Retrieve the Configuration Options

Example

Response Status Codes

Retrieve the Deployment Type

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/job/job-results>

Version: current [26.x]

On this page

Job Results

Use the Job API to retrieve results for a specific job.

Job Results Object

```
{  
  "rowCount": 11,
```

```

"schema": [
  {
    "name": "pickup_datetime",
    "type": {
      "name": "TIMESTAMP"
    }
  },
  {
    "name": "passenger_count",
    "type": {
      "name": "BIGINT"
    }
  },
  {
    "name": "trip_distance_mi",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "fare_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "tip_amount",
    "type": {
      "name": "DOUBLE"
    }
  },
  {
    "name": "total_amount",
    "type": {
      "name": "DOUBLE"
    }
  }
],
"rows": [
  {
    "pickup_datetime": "2013-02-10 20:00:00.000",
    "passenger_count": 9,
    "trip_distance_mi": 3,
    "fare_amount": 10,
    "tip_amount": 3,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:15:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.26,
    "fare_amount": 7.5,
    "tip_amount": 0,
    "total_amount": 8
  }
]

```

```

},
{
  "pickup_datetime": "2013-05-31 16:40:00.000",
  "passenger_count": 1,
  "trip_distance_mi": 0.73,
  "fare_amount": 5,
  "tip_amount": 1.2,
  "total_amount": 7.7
},
{
  "pickup_datetime": "2013-05-27 19:03:00.000",
  "passenger_count": 2,
  "trip_distance_mi": 9.23,
  "fare_amount": 27.5,
  "tip_amount": 5,
  "total_amount": 38.33
},
{
  "pickup_datetime": "2013-05-31 16:24:00.000",
  "passenger_count": 1,
  "trip_distance_mi": 2.27,
  "fare_amount": 12,
  "tip_amount": 0,
  "total_amount": 13.5
},
{
  "pickup_datetime": "2013-05-27 19:17:00.000",
  "passenger_count": 1,
  "trip_distance_mi": 0.71,
  "fare_amount": 5,
  "tip_amount": 0,
  "total_amount": 5.5
},
{
  "pickup_datetime": "2013-05-27 19:11:00.000",
  "passenger_count": 1,
  "trip_distance_mi": 2.52,
  "fare_amount": 10.5,
  "tip_amount": 3.15,
  "total_amount": 14.15
},
{
  "pickup_datetime": "2013-05-31 16:41:00.000",
  "passenger_count": 5,
  "trip_distance_mi": 1.01,
  "fare_amount": 6,
  "tip_amount": 1.1,
  "total_amount": 8.6
},
{
  "pickup_datetime": "2013-05-31 16:37:00.000",
  "passenger_count": 1,
  "trip_distance_mi": 1.25,
  "fare_amount": 8.5,

```

```

    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

Job Results Attributes

rowCount Integer

Number of rows the job returned.

Example: 11

schema Array of Object

Array of schema definitions for the data the job returned.

rows Array of Object

Array of the data the job returned for each row of results.

Retrieve Job Results

Method and URL

```
GET /api/v3/job/{id}/results
```

Parameters

id Path String (UUID)

Unique identifier of the job to retrieve. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

limit Query Integer Optional

Number of rows to retrieve. Maximum valid value is `500`. Default is `100`. Use with the `offset` query parameter to paginate and retrieve more than the first 500 results. Read [Limit and Offset Query Parameters](#) for usage examples.

offset Query Integer Optional

Number of rows to skip for pagination. Default is `0`. Read [Limit and Offset Query Parameters](#) for usage examples.

Example Request

```
curl -X GET 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00/results' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response for a COMPLETED Job

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "fare_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ]
}
```

```

    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ],
  "rows": [
    {
      "pickup_datetime": "2013-02-10 20:00:00.000",
      "passenger_count": 9,
      "trip_distance_mi": 3,
      "fare_amount": 10,
      "tip_amount": 3,
      "total_amount": 13
    },
    {
      "pickup_datetime": "2013-05-27 19:15:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 1.26,
      "fare_amount": 7.5,
      "tip_amount": 0,
      "total_amount": 8
    },
    {
      "pickup_datetime": "2013-05-31 16:40:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 0.73,
      "fare_amount": 5,
      "tip_amount": 1.2,
      "total_amount": 7.7
    },
    {
      "pickup_datetime": "2013-05-27 19:03:00.000",
      "passenger_count": 2,
      "trip_distance_mi": 9.23,
      "fare_amount": 27.5,
      "tip_amount": 5,
      "total_amount": 38.33
    },
    {
      "pickup_datetime": "2013-05-31 16:24:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 2.27,
      "fare_amount": 12,
      "tip_amount": 0,
      "total_amount": 13.5
    },
    {
      "pickup_datetime": "2013-05-27 19:17:00.000",
      "passenger_count": 1,
      "trip_distance_mi": 0.71,
      "fare_amount": 5,

```



```

    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,
    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {
    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

For jobs with a jobState value other than `COMPLETED`, the response includes an error message instead of the job results object. The job's status is listed inside brackets in the error message (`CANCELED`, `FAILED`, or `RUNNING`):

Example Response for a FAILED Job

```

{

```

```
"errorMessage": "Can not fetch details for a job that is in [FAILED] state.",
"moreInfo": ""
}
```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Job](/current/reference/api/job/)

Job Results Attributes

Retrieve Job Results

Parameters

Example Request

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/job/job-results/>

Version: current [26.x]

On this page

Job Results

Use the Job API to retrieve results for a specific job.

Job Results Object

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    }
  ],
}
```

```

{
  "name": "passenger_count",
  "type": {
    "name": "BIGINT"
  }
},
{
  "name": "trip_distance_mi",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "fare_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "tip_amount",
  "type": {
    "name": "DOUBLE"
  }
},
{
  "name": "total_amount",
  "type": {
    "name": "DOUBLE"
  }
}
],
"rows": [
  {
    "pickup_datetime": "2013-02-10 20:00:00.000",
    "passenger_count": 9,
    "trip_distance_mi": 3,
    "fare_amount": 10,
    "tip_amount": 3,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:15:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.26,
    "fare_amount": 7.5,
    "tip_amount": 0,
    "total_amount": 8
  },
  {
    "pickup_datetime": "2013-05-31 16:40:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.73,
    "fare_amount": 5,
    "tip_amount": 1.2,

```

```

    "total_amount": 7.7
  },
  {
    "pickup_datetime": "2013-05-27 19:03:00.000",
    "passenger_count": 2,
    "trip_distance_mi": 9.23,
    "fare_amount": 27.5,
    "tip_amount": 5,
    "total_amount": 38.33
  },
  {
    "pickup_datetime": "2013-05-31 16:24:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.27,
    "fare_amount": 12,
    "tip_amount": 0,
    "total_amount": 13.5
  },
  {
    "pickup_datetime": "2013-05-27 19:17:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.71,
    "fare_amount": 5,
    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,
    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {
    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,

```

```

    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

Job Results Attributes

rowCount Integer

Number of rows the job returned.

Example: 11

schema Array of Object

Array of schema definitions for the data the job returned.

rows Array of Object

Array of the data the job returned for each row of results.

Retrieve Job Results

Method and URL

```
GET /api/v3/job/{id}/results
```

Parameters

id Path String (UUID)

Unique identifier of the job to retrieve. Get the job ID from responses to [SQL API](#) requests.

Example: 6j6c34cf-9drf-b07a-5ab7-abea69a66d00

limit Query Integer Optional

Number of rows to retrieve. Maximum valid value is `500`. Default is `100`. Use with the `offset` query parameter to paginate and retrieve more than the first 500 results. Read [Limit and Offset Query Parameters](#) for usage examples.

offset Query Integer Optional

Number of rows to skip for pagination. Default is `0`. Read [Limit and Offset Query Parameters](#) for usage examples.

Example Request

```
curl -X GET 'https://{hostname}/api/v3/job/6j6c34cf-9drf-b07a-5ab7-abea69a66d00/results' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Example Response for a COMPLETED Job

```
{
  "rowCount": 11,
  "schema": [
    {
      "name": "pickup_datetime",
      "type": {
        "name": "TIMESTAMP"
      }
    },
    {
      "name": "passenger_count",
      "type": {
        "name": "BIGINT"
      }
    },
    {
      "name": "trip_distance_mi",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "fare_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "tip_amount",
      "type": {
        "name": "DOUBLE"
      }
    },
    {
      "name": "total_amount",
      "type": {
        "name": "DOUBLE"
      }
    }
  ]
}
```

```

],
"rows": [
  {
    "pickup_datetime": "2013-02-10 20:00:00.000",
    "passenger_count": 9,
    "trip_distance_mi": 3,
    "fare_amount": 10,
    "tip_amount": 3,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:15:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.26,
    "fare_amount": 7.5,
    "tip_amount": 0,
    "total_amount": 8
  },
  {
    "pickup_datetime": "2013-05-31 16:40:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.73,
    "fare_amount": 5,
    "tip_amount": 1.2,
    "total_amount": 7.7
  },
  {
    "pickup_datetime": "2013-05-27 19:03:00.000",
    "passenger_count": 2,
    "trip_distance_mi": 9.23,
    "fare_amount": 27.5,
    "tip_amount": 5,
    "total_amount": 38.33
  },
  {
    "pickup_datetime": "2013-05-31 16:24:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.27,
    "fare_amount": 12,
    "tip_amount": 0,
    "total_amount": 13.5
  },
  {
    "pickup_datetime": "2013-05-27 19:17:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 0.71,
    "fare_amount": 5,
    "tip_amount": 0,
    "total_amount": 5.5
  },
  {
    "pickup_datetime": "2013-05-27 19:11:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.52,

```

```

    "fare_amount": 10.5,
    "tip_amount": 3.15,
    "total_amount": 14.15
  },
  {
    "pickup_datetime": "2013-05-31 16:41:00.000",
    "passenger_count": 5,
    "trip_distance_mi": 1.01,
    "fare_amount": 6,
    "tip_amount": 1.1,
    "total_amount": 8.6
  },
  {
    "pickup_datetime": "2013-05-31 16:37:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 1.25,
    "fare_amount": 8.5,
    "tip_amount": 0,
    "total_amount": 10
  },
  {
    "pickup_datetime": "2013-05-31 16:39:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 2.04,
    "fare_amount": 10,
    "tip_amount": 1.5,
    "total_amount": 13
  },
  {
    "pickup_datetime": "2013-05-27 19:02:00.000",
    "passenger_count": 1,
    "trip_distance_mi": 11.73,
    "fare_amount": 32.5,
    "tip_amount": 8.12,
    "total_amount": 41.12
  }
]
}

```

For jobs with a jobState value other than `COMPLETED`, the response includes an error message instead of the job results object. The job's status is listed inside brackets in the error message (`CANCELED`, `FAILED`, or `RUNNING`):

Example Response for a FAILED Job

```

{
  "errorMessage": "Can not fetch details for a job that is in [FAILED] state.",
  "moreInfo": ""
}

```

Response Status Codes

200 OK

400 Bad Request

403 Forbidden

404 Not Found

Was this page helpful?

[Previous

Job](/current/reference/api/job/)

Job Results Attributes

Retrieve Job Results

Parameters

Example Request

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/reflections/reflection-recommendations>

Version: current [26.x]

On this page

Recommendations

Use the Recommendations API to get job-based recommendations and get and create usage-based Reflections that can accelerate your queries.

Getting **job-based recommendations** requires making the following two API requests:

Submit the job IDs of jobs that have run SQL queries. These are job IDs of the queries for which you want to retrieve recommendations in further requests. This request returns the job ID to use in the second request.

Retrieve job-based recommendations for Reflections that can accelerate your queries. Use the job ID that was returned in your first request to make the request for recommendations.

Creating Reflections from **usage-based recommendations** requires making the following two API requests:

Retrieve usage-based recommendations for Reflections. This request returns the parameters to use in the body of the second request.

Create Reflections from usage-based recommendations that can accelerate your queries from the usage-based recommendations. Use the recommendation ID and Reflection request body that were returned in your first request to create the

Reflections.

caution

The ``POST /api/v3/reflection/recommendations`` endpoint is deprecated. In its place, use the job-based and usage-based endpoints described on this page to retrieve and refresh Reflection recommendations.

Recommendation Object (Raw Reflection)

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "azure_3",
          "table_2"
        ],
        "sql": "--Default Raw Reflection"
      },
      "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
      "reflectionRequestBody": {
        "type": "RAW",
        "name": "raw_47f54460-543f-430f-a9e5-ca71d246265e",
        "datasetId": "45b9d98b-b0dc-4dd2-a271-d971ae998c0c",
        "enabled": true,
        "arrowCachingEnabled": false,
        "dimensionFields": [],
        "measureFields": [],
        "displayFields": [
          {
            "name": "passenger_count"
          },
          {
            "name": "EXPR$1"
          }
        ],
        "entityType": "reflection"
      },
      "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
      "jobIds": [
        "13ffb629-9f0e-4265-97df-99bf0d425813"
      ],
      "jobCount": 1,
      "recommendationId": "9be8a451-4190-4618-a72e-9932f790c744",
      "reflectionScore": 50.67,
      "avgImprovementFactor": 10.43,
      "avgImprovementMs": 7196
    }
  ],
  "canAlterReflections": true
}
```

Recommendation Object (Aggregation Reflection)

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "recommended_view",
          "view_1"
        ],
        "sql": "SELECT * FROM Samples.samples.dremio.com.\"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC"
      },
      "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
      "reflectionRequestBody": {
        "type": "AGGREGATION",
        "name": "agg_0e0c4ab9-def7-48da-81f1-ca8c1dalled4",
        "datasetId": "2df93b5a-eb46-4687-8460-b61e471d20ef",
        "enabled": true,
        "arrowCachingEnabled": false,
        "dimensionFields": [
          {
            "name": "passenger_count",
            "granularity": "DATE"
          }
        ],
        "measureFields": [
          {
            "name": "fare_amount",
            "measureTypeList": [
              "SUM",
              "COUNT"
            ]
          }
        ],
        "displayFields": [],
        "entityType": "reflection"
      },
      "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
      "jobIds": [
        "1ded81f8-4d06-4d09-8163-9e2517027d8d"
      ],
      "jobCount": 1,
      "recommendationId": "1855d2dd-4106-4359-a97a-e08a916096e6",
      "reflectionScore": 60.12,
      "avgImprovementFactor": 8.39,
      "avgImprovementMs": 5400
    }
  ],
  "canAlterReflections": true
}
```

Recommendation Attributes

data Array of Object

List of recommended Reflection objects for the submitted job IDs.

canAlterReflections Boolean

If the columns in the recommended Reflection can be edited, added, and removed, `true`. Otherwise, `false`.

Example: true

Attributes of Objects in the `data` Array

viewRequestBody Object

The fields to include in a request to the Catalog API to create the view on which to define the recommended Reflection.

viewRequestEndpoint String

The endpoint to use when submitting a request to the Catalog API to create the view on which to define the recommended Reflection.

reflectionRequestBody Object

The fields to include in a request to the Reflection API to create the recommended Reflection.

reflectionRequestEndpoint String

The endpoint to use when submitting the request to the Reflection API to create the recommended Reflection.

jobIds Array of String

The job IDs of the queries for which the Reflection recommendations are given.

Example: ["13ffb629-9f0e-4265-97df-99bf0d425813"]

jobCount Array of String

The number of jobs for which Reflection recommendations are given.

Example: 1

recommendationId Array of String

The ID of the recommended Reflection.

Example: ["9be8a451-4190-4618-a72e-9932f790c744"]

reflectionScore Double

Score for the recommended Reflection's quality, on a scale of 0 (worst) to 100 (best). The reflectionScore value considers the recommended Reflection's anticipated quality

compared to existing Reflections and other recommended Reflections, as well as the likely improvement in query run times if the recommended Reflection is implemented.

Example: 50.67

avgImprovementFactor Double

The likely average multiplicative rate of improvement for each query if you implement the recommended Reflection. For example, if the avgImprovementFactor value is 2.34, implementing the recommended Reflection is likely to speed up each query by 2.34 times, on average.

Example: 10.43

avgImprovementMs Double

The likely average improvement, in milliseconds, for each query if you implement the recommended Reflection. For example, if the avgImprovementMs value is 5400, implementing the recommended Reflection is likely to save an average of 5400 milliseconds for each query that uses the Reflection.

Example: 7196

Attributes of the `viewRequestBody` Object

entityType String

Type of catalog entity. For views, the entityType is `dataset`.

type String

Type of dataset. For views, the type is `VIRTUAL_DATASET`.

path Array of String

Path to the location where the view should be created within Dremio, expressed in an array. The path lists each level of hierarchy in order, from outer to inner: Arctic source or catalog first, then folder and subfolders, then a name for the view itself as the last item in the array. Views can only be created in Arctic sources and the project's Arctic catalog.

Example: ["azure_3","table_2"]

sql String

For aggregation Reflections, the SQL query to use to create the view. For default raw Reflections, the sql value `Default Raw Reflection`: creating a view is unnecessary because raw recommendations are given only for existing views.

Attributes of the `reflectionRequestBody` Object

type String

Reflection type. For details, read Types of Reflections.

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio console, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: raw_47f54460-543f-430f-a9e5-ca71d246265e

datasetId String

Unique identifier for the anchor dataset to associate with the Reflection.

Example: 45b9d98b-b0dc-4dd2-a271-d971ae998c0c

enabled Boolean

If the Reflection is available for accelerating queries, `true`. Otherwise, `false`.

Example: true

arrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, `true`. Otherwise, `false`.

Example: false

dimensionFields Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. Dimension fields are the fields you expect to group by when analyzing data. Valid only for aggregation Reflections. For raw Reflections or if the anchor dataset does not include any dimension fields, the dimensionFields value is an empty array. For aggregation Reflections, if the anchor dataset includes dimension fields, each object in the dimensionFields array contains two attributes: name and granularity.

Example: [{"name": "pickup_date", "granularity": "DATE"}, {"name": "pickup_datetime", "granularity": "DATE"}, {"name": "dropoff_date", "granularity": "DATE"}, {"name": "dropoff_datetime", "granularity": "DATE"}, {"name": "passenger_count", "granularity": "DATE"}, {"name": "total_amount", "granularity": "DATE"}]

measureFields Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. Measure fields are the fields you expect to use for calculations when analyzing the data. Valid only for aggregation Reflections. For raw Reflections or if the anchor dataset does not include any measure fields, the measureFields value is an empty array. For aggregation Reflections, if the anchor dataset includes measure fields, each object in the measureFields array contains two attributes: name and measureTypeList.

Example: [{"name": "passenger_count", "measureTypeList": ["SUM", "COUNT"]}, {"name": "trip_distance_mi", "measureTypeList": ["SUM", "COUNT"]}, {"name": "fare_amount", "measureTypeList": ["SUM", "COUNT"]}, {"name": "surcharge", "measureTypeList": ["SUM", "COUNT"]}, {"name": "tip_amount", "measureTypeList": ["SUM", "COUNT"]}]

```
[{"SUM","COUNT"}],{"name": "total_amount","measureTypeList": ["SUM","COUNT"]}]
```

displayFields Array of Object

Information about the fields displayed from the anchor dataset. Valid only for raw Reflections. For aggregation Reflections or if the anchor dataset does not include any display fields, the value is an empty array. For raw Reflections, if the anchor dataset includes display fields, each object in the displayFields array contains one attribute: name.

Example: [{"name": "passenger_count"}, {"name": "EXPR\$1"}]

entityType String

Type of entity. For Reflection objects, the entityType is `reflection`.

Submit Job IDs

Submit the job IDs of queries for which you want to request Reflection recommendations.

The response includes objects that contain an id attribute and value for each job ID you submit. Use these id values to retrieve recommendations for Reflections to accelerate the queries.

Method and URL

```
POST /api/v3/reflection/recommendations/job-based/
```

Parameters

jobIds Body Array of String

The job IDs of the queries for which you want to request Reflection recommendations. To get the job IDs, use the SQL API or find them on the Jobs page in the Dremio console. Use a comma-separated list to submit multiple job IDs.

Example: ["a7efcd50-791a-48e8-bb05-391b4411e66b"]

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations/job-based/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "jobIds":
  ["a7efcd50-791a-48e8-bb05-391b4411e66b", "c2485882-e6b7-4aa8-af5b-a825d2870589"]
}'
```

Response

```
{
  "id": "13ffb629-9f0e-4265-97df-99bf0d425813"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

405 Method Not Allowed

500 Internal Server Error

Retrieve Job-Based Recommendations

Retrieve job-based recommended Reflections to accelerate the queries whose job IDs you submitted.

For default raw Reflections, each recommendation comprises the path to the view on which to define the Reflection and the parameters to use in a request to create the Reflection.

For aggregation Reflections, each recommendation comprises the parameters to use in a request to create a view on which to define the recommended Reflection and the parameters to use in a request to create the Reflection.

After you retrieve the recommended Reflections for your queries, use the Catalog API to create the recommended views. Then, use the Reflection API to create the desired Reflections.

note

Before submitting Catalog API requests to create the recommended views for aggregation Reflections, create a folder named ``recommended_view``. In your Catalog API requests, the ``path`` parameter must include the full path to the ``recommended_view`` folder. If you prefer to use a different folder name, replace ``recommended_view`` with your folder name in the ``path`` parameter when making the Catalog API request.

Method and URL

```
GET /api/v3/reflection/recommendations/job-based/{id}/results/
```

Parameters

id Path String

The id value returned in the response to your request to submit the job ID or IDs of the

queries for which you want to retrieve recommended Reflections.

Example: 13ffb629-9f0e-4265-97df-99bf0d425813

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/reflection/recommendations/job-based/13ffb629-9f0e-4265-97df-99bf0d425813/results/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "azure_3",
          "table_2"
        ],
        "sql": "--Default Raw Reflection"
      },
      "viewRequestEndpoint": "{hostname}/api/v3/catalog",
      "reflectionRequestBody": {
        "type": "RAW",
        "name": "raw_47f54460-543f-430f-a9e5-ca71d246265e",
        "datasetId": "45b9d98b-b0dc-4dd2-a271-d971ae998c0c",
        "enabled": true,
        "arrowCachingEnabled": false,
        "dimensionFields": [],
        "measureFields": [],
        "displayFields": [
          {
            "name": "passenger_count"
          },
          {
            "name": "EXPR$1"
          }
        ],
        "entityType": "reflection"
      },
      "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
      "jobIds": [
        "13ffb629-9f0e-4265-97df-99bf0d425813"
      ],
      "jobCount": 1,
      "recommendationId": "9be8a451-4190-4618-a72e-9932f790c744",
    }
  ]
}
```

```

    "reflectionScore": 50.67,
    "avgImprovementFactor": 10.43,
    "avgImprovementMs": 7196
  }
],
"canAlterReflections": true
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

405 Method Not Allowed

500 Internal Server Error

Retrieve Usage-Based Recommendations

Retrieve usage-based Reflection recommendations. The response includes the `reflectionRequestBody` and `recommendationId` attributes to use as body parameters in your request to [create usage-based Reflections](#).

Method and URL

```
GET /api/v3/reflection/recommendations/usage-based/
```

Example

Request

```

curl -X GET 'https://{hostname}/api/v3/reflection/recommendations/usage-based/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'

```

Response

```

{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "type": "VIRTUAL_DATASET",
        "path": [
          "prodFolder",
          "cost_based"
        ],

```

```

    "sql": "--Default Raw Reflection"
  },
  "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
  "reflectionRequestBody": {
    "type": "RAW",
    "name": "AutoRef_cost_based_raw",
    "datasetId": "61d689a2-cd04-4d5d-84a7-021bdc15bff6",
    "enabled": true,
    "arrowCachingEnabled": false,
    "dimensionFields": [],
    "measureFields": [],
    "displayFields": [
      {
        "name": "passenger_count"
      },
      {
        "name": "pickup_datetime"
      },
      {
        "name": "EXPR$2"
      }
    ],
    "entityType": "reflection"
  },
  "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
  "jobIds": [
    "1975ec43-349a-9310-2e40-acbd8d025c00",
    "1975ac7c-6541-86db-ae43-dcef2ffee300",
    "1974b533-8c88-946b-92ce-ee5ab7791500"
  ],
  "jobCount": 3,
  "recommendationId": "prodFolder.cost_based",
  "reflectionScore": 36.928031592652964,
  "avgImprovementFactor": 10.00000020692081,
  "avgImprovementMs": 7393.800016999235
}
],
"canAlterReflections": true
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

500 Internal Server Error

Create Reflections from Usage-Based Recommendations

Create Reflections to accelerate queries using the usage-based recommendations that you retrieved.

note

You must retrieve usage-based recommendations to get the parameters you need for this request.

Method and URL

```
POST /api/v3/reflection/recommendations/usage-based/
```

Parameters

reflection Body Object

Information about the usage-based Reflection to create. The Reflection object includes the contents of the reflectionRequestBody included in the response for requests to retrieve usage-based recommendations.

NOTE: If desired, you may change the name of the Reflection by changing the value for the Reflection.name parameter in the body of your request. Dremio ignores any changes to the values of other parameters in the Reflection object.

recommendationId Body String

Identifier for the usage-based recommendation you want to use to create Reflections. The recommendationId is included in the response for requests to retrieve usage-based recommendations.

Example: prodFolder.cost_based

Parameters of the `reflection` Object

type Body String

Reflection type. Value must be `RAW`.

name Body String

User-provided name for the Reflection.

NOTE: If desired, you may change the name of the Reflection by changing the value for the name parameter in the body of your request.

Example: AutoRef_cost_based_raw

datasetId Body String

Unique identifier for the anchor dataset to associate with the Reflection.

Example: 61d689a2-cd04-4d5d-84a7-021bdc15bff6

enabled Body Boolean

If the Reflection is available for accelerating queries, `true`. Otherwise, `false`.

Example: true

arrowCachingEnabled Body Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, `true`. Otherwise, `false`.

Example: false

dimensionFields Body Array of Object

Information about the dimension fields from the anchor dataset used in the Reflection. For raw Reflections, the dimensionFields value is an empty array.

Example: []

measureFields Body Array of Object

Information about the measure fields from the anchor dataset used in the Reflection. For raw Reflections, the measureFields value is an empty array.

Example: []

displayFields Body Array of Object

Information about the fields displayed from the anchor dataset. Valid only for raw Reflections. If the anchor dataset includes display fields, each object in the displayFields array contains one attribute: name.

Example: [{"name": "passenger_count"}, {"name": "EXPR\$1"}]

entityType Body String

Type of entity. For Reflection objects, the entityType is `reflection`.

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations/usage-based/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "reflection": {
    "type": "RAW",
    "name": "AutoRef_cost_based_raw",
    "datasetId": "61d689a2-cd04-4d5d-84a7-021bdc15bff6",
    "enabled": true,
    "arrowCachingEnabled": false,
    "dimensionFields": [],
    "measureFields": [],
    "displayFields": [
      {
        "name": "passenger_count"
      }
    ]
  }
}
```

```

    {
      "name": "pickup_datetime"
    },
    {
      "name": "EXPR$2"
    }
  ],
  "entityType": "reflection"
},
"recommendationId": "prodFolder.cost_based"
}

```

Response

```

{
  "id": "c929b8d2-82bf-4175-9476-010ba17c4f7f",
  "type": "RAW",
  "name": "AutoRef_cost_based_raw",
  "tag": "4p/C0EkSud7=",
  "createdAt": "2024-06-28T19:30:30.977Z",
  "updatedAt": "2024-06-28T19:30:30.977Z",
  "datasetId": "61d689a2-cd04-4d5d-84a7-021bdc15bff6",
  "currentSizeBytes": 0,
  "totalSizeBytes": 0,
  "enabled": true,
  "arrowCachingEnabled": false,
  "status": {
    "config": "OK",
    "refresh": "SCHEDULED",
    "availability": "NONE",
    "combinedStatus": "CANNOT_ACCELERATE_SCHEDULED",
    "failureCount": 0,
    "lastDataFetch": "1969-12-31T23:59:59.999Z",
    "expiresAt": "1969-12-31T23:59:59.999Z"
  },
  "displayFields": [
    {
      "name": "passenger_count"
    },
    {
      "name": "pickup_datetime"
    },
    {
      "name": "EXPR$2"
    }
  ],
  "partitionDistributionStrategy": "CONSOLIDATED",
  "canView": true,
  "canAlter": true,
  "entityType": "reflection"
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

500 Internal Server Error

Refresh Usage-Based Recommendations

Process collected data about view usage, clear existing usage-based recommendations, and generate new usage-based recommendations.

Use the usage-based endpoints to [retrieve](#) and [create](#) Reflections based on the refreshed recommendations this endpoint creates.

Method and URL

```
POST /api/v3/reflection/recommendations/usage-based/refresh/
```

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations/usage-based/refresh/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

This endpoint returns an empty response body with a `202 Accepted` response status code. Dremio updates the recommendations asynchronously, so it may take several minutes before you can [retrieve](#) the updated recommendations.

Response Status Codes

202 Accepted

400 Bad Request

401 Unauthorized

500 Internal Server Error

Delete Usage-Based Recommendations

Delete all collected usage data and all current Reflection recommendations.

note

We recommend deleting recommendations only when troubleshooting.

Method and URL

```
DELETE /api/v3/reflection/recommendations/usage-based/
```

Example Request

```
curl -X DELETE 'https://{hostname}/api/v3/reflection/recommendations/usage-based/' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

This endpoint returns an empty response body with a `202 Accepted` response status code. Dremio deletes the recommendations asynchronously, so it may take several minutes for the deletion to complete.

Response Status Codes

202 Accepted

400 Bad Request

401 Unauthorized

500 Internal Server Error

Retrieve Recommendations (Deprecated)"

caution

The `POST /api/v3/reflection/recommendations` endpoint described in this section is deprecated. In its place, use the `job-based` and `usage-based` endpoints to retrieve and refresh Reflection recommendations.

Use the Recommendations API to submit job IDs of jobs that ran SQL queries, and receive recommendations for aggregation Reflections that can accelerate those queries.

For more information, see [Sending Requests to the Recommendations API](#).

Recommendation Object

```
{
  "data": [
    {
      "viewRequestBody": {
        "entityType": "dataset",
        "path": [
          "recommended_view",
          "Dataset_be919a56-f18b-421b-9612-711a1cc51b69"
        ],
        "type": "VIRTUAL_DATASET",
        "sql": "SELECT * FROM Samples.samples.dremio.com.\"NYC-taxi-trips\" WHERE
trip_distance_mi <= 2.0 ORDER BY trip_distance_mi ASC",
```



```

    }
    "viewRequestEndpoint": "POST {hostname}/api/v3/catalog",
    "reflectionRequestBody": {
      "type": "AGGREGATION",
      "name": "agg_250e70d1-5e2a-4938-a1a1-95f664085099",
      "datasetId": "be919a56-f18b-421b-9612-711a1cc51b69",
      "enabled": true,
      "dimensionFields": [
        {
          "name": "passenger_count",
          "granularity": "DATE"
        }
      ],
      "measureFields": [
        {
          "name": "fare_amount",
          "measureTypeList": [
            "SUM",
            "COUNT"
          ]
        }
      ],
      "entityType": "reflection"
    }
    "reflectionRequestEndpoint": "POST {hostname}/api/v3/reflection",
    "jobIds": ["6j6c34cf-9drf-b07a-5ab7-abea69a66d00"]
  }
],
"canAlterReflections": true
}

```

Recommendation Attributes (Deprecated)"

viewRequestBody Array of Object

The fields that you can include in a request to the Catalog API for creating the view on which to define the recommended aggregation Reflection.

For descriptions of these fields, see [View](#).

viewRequestEndpoint String

The endpoint to use when submitting the request to the Catalog API to create the view on which to define the Reflection.

reflectionRequestBody Array of Object

The fields that you can include in a request to the Reflection API for creating the recommended aggregation Reflection.

For descriptions of these fields, see [Reflection](#).

reflectionRequestEndpoint String

The endpoint to use when submitting the request to the Reflection API to create the aggregation Reflection.

jobIds Array of String

The IDs of the jobs that ran the queries for which the recommendation is given.

canAlterReflections String

Indicates whether the columns in the Reflection can be edited, and whether columns can be added or removed.

Request Recommendations (Deprecated)"

caution

This endpoint is deprecated. In its place, use the [job-based](#) and [usage-based](#) endpoints to retrieve and refresh Reflection recommendations.

Request recommended aggregation Reflections to accelerate the queries associated with the provided job ID or IDs.

Method and URL

```
POST /api/v3/reflection/recommendations
```

Parameters (Deprecated)"

jobIds Body Array of String

A list of the job IDs of jobs that have run the SQL commands that you want to receive one or more recommended Reflections for.

Example Request

```
curl -X POST 'https://{hostname}/api/v3/reflection/recommendations' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data
'{
  "jobIds": [
    "1a515250-7572-0f9b-f5e5-89f505b55200",
    "1a515292-583c-e407-79ef-9f9b494fa600"
  ]
}'
```

Was this page helpful?

[Previous

Reflection](/current/reference/api/reflections/)[Next

Reflection Summary](/current/reference/api/reflections/reflection-summary)

Recommendation Attributes

Submit Job IDs

Parameters

Example

Response Status Codes

Retrieve Job-Based Recommendations

Parameters

Example

Response Status Codes

Retrieve Usage-Based Recommendations

Example

Response Status Codes

Create Reflections from Usage-Based Recommendations

Parameters

Example

Response Status Codes

Refresh Usage-Based Recommendations

Example Request

Response Status Codes

Delete Usage-Based Recommendations

Example Request

Response Status Codes

Retrieve Recommendations (Deprecated)

Example Request

Source:
<https://docs.dremio.com/current/reference/api/reflections/reflection-summary>

Version: current [26.x]

On this page

Reflection Summary Enterprise

Use the Reflection API to retrieve a Reflection summary that includes the raw and aggregation Reflections for the Dremio instance.

Reflection summary objects are different from Reflection objects. Reflection summaries do not include certain attributes that define the Reflection, like the display, dimension, measure, sort, and partition attributes. Reflection summaries do include several attributes that do not appear in Reflection objects, like `datasetType`, `datasetPath`, and counts and links for considered, matched, and chosen jobs.

Reflection Summary Object

```
{
  "data": [
    {
      "createdAt": "2022-07-05T19:19:40.244Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
      "reflectionType": "AGGREGATION",
      "name": "NYC_taxi_agg",
      "currentSizeBytes": 9272,
      "outputRecords": 51,
      "totalSizeBytes": 9272,
      "datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
      "datasetType": "VIRTUAL_DATASET",
      "datasetPath": [
        "Samples",
        "samples.dremio.com",
        "NYC-taxi-trips"
      ],
      "status": {
        "configStatus": "OK",
        "refreshStatus": "MANUAL",
        "availabilityStatus": "AVAILABLE",
        "combinedStatus": "CAN_ACCELERATE",
        "refreshMethod": "FULL",
        "failureCount": 0,
        "lastDataFetchAt": "2023-01-13T19:05:03.532Z",
        "expiresAt": "3022-05-16T19:46:02.342Z",
        "lastRefreshDurationMillis": 46387
      },
      "consideredCount": 202,
      "matchedCount": 45,
      "chosenCount": 5,
      "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
      "isArrowCachingEnabled": false,
    }
  ]
}
```

```

    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-26T23:27:04.281Z",
    "updatedAt": "2023-01-26T23:27:04.281Z",
    "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
    "reflectionType": "RAW",
    "name": "Raw Reflection",
    "currentSizeBytes": 0,
    "outputRecords": -1,
    "totalSizeBytes": 0,
    "datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
    "datasetType": "PHYSICAL_DATASET",
    "datasetPath": [
      "@dremio",
      "test"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "GIVEN_UP",
      "availabilityStatus": "NONE",
      "combinedStatus": "FAILED",
      "refreshMethod": "NONE",
      "failureCount": 3,
      "lastFailureMessage": "The Default engine is not online.",
      "lastDataFetchAt": null,
      "expiresAt": null,
      "lastRefreshDurationMillis": -1
    },
    "consideredCount": 0,
    "matchedCount": 0,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:19.030Z",
    "updatedAt": "2023-01-13T19:50:19.030Z",
    "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
    "reflectionType": "AGGREGATION",
    "name": "dataset991_agg991",

```

```

"currentSizeBytes": 9273,
"outputRecords": 51,
"totalSizeBytes": 9273,
"datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
"datasetType": "VIRTUAL_DATASET",
"datasetPath": [
  "temp",
  "dataset991"
],
"status": {
  "configStatus": "OK",
  "refreshStatus": "MANUAL",
  "availabilityStatus": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "refreshMethod": "FULL",
  "failureCount": 0,
  "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
  "expiresAt": "3022-05-16T19:46:02.342Z",
  "lastRefreshDurationMillis": 11697
},
"consideredCount": 60,
"matchedCount": 9,
"chosenCount": 0,
"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"isArrowCachingEnabled": false,
"isCanView": true,
"isCanAlter": true,
"isEnabled": true
},
{
  "createdAt": "2023-01-13T19:50:17.714Z",
  "updatedAt": "2023-01-13T19:50:17.714Z",
  "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
  "reflectionType": "RAW",
  "name": "dataset991_raw991",
  "currentSizeBytes": 818790,
  "outputRecords": 29467,
  "totalSizeBytes": 818790,
  "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
  "datasetType": "VIRTUAL_DATASET",
  "datasetPath": [
    "temp",
    "dataset991"
  ],
  "status": {
    "configStatus": "OK",

```

```

    "refreshStatus": "MANUAL",
    "availabilityStatus": "AVAILABLE",
    "combinedStatus": "CAN_ACCELERATE",
    "refreshMethod": "FULL",
    "failureCount": 0,
    "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
    "expiresAt": "3022-05-16T19:46:02.747Z",
    "lastRefreshDurationMillis": 16666
  },
  "consideredCount": 54,
  "matchedCount": 37,
  "chosenCount": 0,
  "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
}
],
"nextPageToken": "CiQxNjc0MjhIYi03OTM2LTRlYtItYTFmYi0yM2IxYWM2ZTk0NTQSAA==",
"isCanAlterReflections": true
}

```

Reflection Summary Attributes

data Array of Object

List of Reflection-summary objects for each Reflection in the Dremio instance.

nextPageToken String

Opaque string to pass for the `pageToken` query parameter in the next request to retrieve the next set of results. If nextPageToken is not included in the response, all available resources have been returned.

Example: CiQxNjc0MjhIYi03OTM2LTRlYtItYTFmYi0yM2IxYWM2ZTk0NTQSAA==

isCanAlterReflections Boolean

If the current user has project-level privileges to alter Reflections, the value is `true`. Otherwise, the value is `false`.

Example: true

Attributes of Objects in the `data` Array

createdAt String

Date and time that the Reflection was created, in UTC format.

Example: 2022-07-05T19:19:40.244Z

updatedAt String

Date and time that the Reflection was last updated, in UTC format.

Example: 2023-01-13T19:46:01.313Z

id String (UUID)

Unique identifier of the Reflection.

Example: 27077c03-ae49-454c-a7bb-a9a8b5eca224

reflectionType String

Reflection type. For more information, read [Types of Reflections](#).

Enum: RAW, AGGREGATION

Example: AGGREGATION

name String

User-provided name for the Reflection. For Reflections created in the Dremio UI, if the user did not provide a name, the default values are `Raw Reflection` and `Aggregation Reflection` (automatically assigned based on the Reflection type).

Example: NYC_taxi_agg

currentSizeBytes Integer

Data size of the latest Reflection job (if one exists), in bytes.

Example: 9272

outputRecords Integer

Number of records returned for the latest Reflection.

Example: 51

totalSizeBytes Integer

Data size of all Reflection jobs that have not been pruned (if any exist), in bytes.

Example: 9272

datasetId String

Unique identifier of the anchor dataset that is associated with the Reflection.

Example: fa7c487f-9550-474e-8a41-4826564c6b09

datasetType String

Type for the anchor dataset that is associated with the Reflection. If the anchor dataset is a table, the type is `PHYSICAL_DATASET`. If the anchor dataset is a view, the type is `VIRTUAL_DATASET`.

Enum: PHYSICAL_DATASET, VIRTUAL_DATASET

Example: VIRTUAL_DATASET

datasetPath String

Path to the anchor dataset that is associated with the Reflection within Dremio, expressed in an array. The path consists of the source or space, followed by any folder and subfolders, followed by the name of the dataset itself as the last item in the array.

Example: ["Samples","samples.dremio.com","NYC Taxi Trips"]

status Object

Information about the status of the Reflection.

Example: {\n "configStatus": "OK",\n "refreshStatus": "MANUAL",\n "availabilityStatus": "AVAILABLE",\n "combinedStatus": "CAN_ACCELERATE",\n "refreshMethod": "FULL",\n "failureCount": 0,\n "lastDataFetchAt": "2023-01-13T19:05:03.532Z",\n "expiresAt": "3022-05-16T19:46:02.342Z",\n "lastRefreshDurationMillis": 46387\n }

consideredCount Integer

Number of jobs that considered the Reflection during planning.

Example: 202

matchedCount Integer

Number of jobs that matched the Reflection during planning.

Example: 45

chosenCount Integer

Number of jobs accelerated by the Reflection.

Example: 5

consideredJobsLink String

Link to list of considered jobs for the Reflection.

Example:

/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D

matchedJobsLink String

Link to list of matched jobs for the Reflection.

Example:

/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATI

ON%22%5D%7D

chosenJobsLink String

Link to list of chosen jobs for the Reflection.

Example:

/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATI
ON%22%5D%7D

isArrowCachingEnabled Boolean

If Dremio converts data from the Reflection's Parquet files to Apache Arrow format when copying that data to executor nodes, the value is ``true``. Otherwise, the value is ``false``.

Example: false

isCanView Boolean

If you can view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

isCanAlter Boolean

If you can create, edit, and view Reflections on all datasets of a source, system, space, or folder, the value is ``true``. Otherwise, the value is ``false``.

Example: true

isEnabled Boolean

If the Reflection is available for accelerating queries, the value is ``true``. Otherwise, the value is ``false``.

Example: true

Attributes of the ``status`` Object

configStatus String

Status of the Reflection configuration. If the value is ``OK``, the Reflection configuration is free of errors. If the value is ``INVALID``, the Reflection configuration contains one or more errors.

Enum: OK, INVALID

Example: OK

refreshStatus String

Status of the Reflection refresh.

``GIVEN_UP``: Dremio attempted to refresh the Reflection multiple times, but each

attempt has failed and Dremio will not make further attempts.

``MANUAL``: Refresh period is set to 0, so you must use the Dremio UI to manually refresh the Reflection.

``RUNNING``: Dremio is currently refreshing the Reflection.

``SCHEDULED``: The Reflection refreshes according to a schedule.

``ON_DATA_CHANGES``: All of the Reflection's underlying tables are in Iceberg format, and the Reflection refreshes automatically if new snapshots are created after an update to the underlying tables.

Enum: `GIVEN_UP`, `MANUAL`, `RUNNING`, `SCHEDULED`, `ON_DATA_CHANGES`

Example: `MANUAL`

availabilityStatus String

Status of the Reflection's availability for accelerating queries.

Enum: `NONE`, `EXPIRED`, `AVAILABLE`

Example: `AVAILABLE`

combinedStatus String

Status of the Reflection based on a combination of config, refresh, and availability.

``CAN_ACCELERATE``: The Reflection is fully functional.

``CAN_ACCELERATE_WITH_FAILURES``: The most recent refresh failed to obtain a status, but Dremio still has a valid materialization.

``CANNOT_ACCELERATE_INITIALIZING``: The Reflection is currently being loaded into the materialization cache. During this time, the Reflection is unable to accelerate queries.

``CANNOT_ACCELERATE_MANUAL``: The Reflection is unable to accelerate any queries, and the ``Never Refresh`` option is selected for the refresh policy.

``CANNOT_ACCELERATE_SCHEDULED``: The Reflection is currently unable to accelerate any queries, but it has been scheduled for a refresh at a future time.

``DISABLED``: The Reflection has been manually disabled.

``EXPIRED``: The Reflection has expired and cannot be used.

``FAILED``: The attempt to refresh the Reflection has failed, typically three times in a row. The Reflection is still usable.

``INVALID``: The Reflection is invalid because the underlying dataset has changed.

``REFRESHING``: The Reflection is currently being refreshed.

Example: `CAN_ACCELERATE`

refreshMethod String

The method used for the most recent refresh of the Reflection. For new Reflections, the

value is `NONE` until planned. For more information, read [Refreshing Reflections](#).

Enum: NONE, FULL, INCREMENTAL

Example: FULL

failureCount Integer

Number of times that an attempt to refresh the Reflection failed.

Example: 0

lastFailureMessage String

The error message from the last failed Reflection refresh. If the refresh of a Reflection never fails or succeeds after a failure, this attribute does not appear.

Example: "The Default engine is not online."

lastDataFetchAt String

Date and time that the Reflection data was last refreshed, in UTC format. If the Reflection is running, failing, or disabled, the lastDataFetchAt value is `1969-12-31T23:59:59.999Z`.

Example: 2023-01-13T19:05:03.532Z

expiresAt String

Date and time that the Reflection expires, in UTC format. If the Reflection is running, failing, or disabled, the expiresAt value is `1969-12-31T23:59:59.999Z`.

Example: 3022-05-16T19:46:02.342Z

lastRefreshDurationMillis Integer

Duration of the most recent refresh for the Reflection. In milliseconds.

Example: 46387

Retrieve a Reflection Summary

Retrieve a summary of the raw and aggregation Reflections in the Dremio instance.

Method and URL

```
GET /api/v3/reflection-summary
```

Parameters

pageToken Query String Optional

Token for retrieving the next page of Reflection summary results. If the Dremio instance has more Reflection summary results than the maximum per page (default 50), the response includes a nextPageToken after the data array. Use the nextPageToken value

in your request URL as the `pageToken` value. Do not change any other query parameters included in the request URL when you use `pageToken`. For more information, read [pageToken Query Parameter](#).

`maxResults` Query Integer Optional

Maximum number of Reflection summaries to return in the response. Maximum valid value is ``100``. Default is ``50``. For more information, read [maxResults Query Parameter](#).

`filter` Query Object Optional

Filters for Reflection name, dataset name, availability status, and refresh status. Value is a URL-encoded string that represents a JSON object. The JSON object specifies the attributes to filter on and the values to match for each attribute. Available filter attributes:

`reflectionType`: ``RAW``, ``AGGREGATION`` (array of string)

`refreshStatus`: ``GIVEN_UP``, ``MANUAL``, ``RUNNING``, ``SCHEDULED``, ``ON_DATA_CHANGES`` (array of string)

`availabilityStatus`: ``NONE``, ``EXPIRED``, ``AVAILABLE`` (array of string)

`configStatus`: ``OK``, ``INVALID`` (array of string)

`enabledFlag`: ``true``, ``false`` (Boolean)

`reflectionNameOrDatasetPath`: full or partial Reflection name or dataset path; case insensitive (string)

`reflectionIds`: IDs of Reflections to retrieve (array of string); must be used alone, with no other filters or query parameters

For more information, read [filter Query Parameter](#).

`orderBy` Query String Optional

Organize the response in ascending (default) or descending order by `reflectionName`, `datasetName`, or `reflectionType`. To specify descending order, precede the `orderBy` value with a ``-`` character. For more information, read [orderBy Query Parameter](#).

Example Request Without Query Parameters

```
curl -X GET 'https://{hostname}/api/v3/reflection-summary'
--header 'Authorization: Bearer <dremioAccessToken>' \
--header "Content-Type: application/json"
```

Example Response

```
{
  "data": [
    {
      "createdAt": "2023-01-13T19:46:01.313Z",
      "updatedAt": "2023-01-13T19:46:01.313Z",
      "id": "27077c03-ae49-454c-a7bb-a9a8b5eca224",
    }
  ]
}
```

```

"reflectionType": "AGGREGATION",
"name": "NYC_taxi_agg",
"currentSizeBytes": 9272,
"outputRecords": 51,
"totalSizeBytes": 9272,
"datasetId": "fa7c487f-9550-474e-8a41-4826564c6b09",
"datasetType": "VIRTUAL_DATASET",
"datasetPath": [
  "Samples",
  "samples.dremio.com",
  "NYC-taxi-trips"
],
"status": {
  "configStatus": "OK",
  "refreshStatus": "MANUAL",
  "availabilityStatus": "AVAILABLE",
  "combinedStatus": "CAN_ACCELERATE",
  "refreshMethod": "FULL",
  "failureCount": 0,
  "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
  "expiresAt": "3022-05-16T19:46:02.342Z",
  "lastRefreshDurationMillis": 46387
},
"consideredCount": 202,
"matchedCount": 45,
"chosenCount": 5,

"consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22422ace5b8a9a-bb7a-c454-94ea-30c77072%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
"isArrowCachingEnabled": false,
"isCanView": true,
"isCanAlter": true,
"isEnabled": true
},
{
  "createdAt": "2023-01-26T23:27:04.281Z",
  "updatedAt": "2023-01-26T23:27:04.281Z",
  "id": "0e3d765a-2291-4a04-81eb-2daf5477cc7d",
  "reflectionType": "RAW",
  "name": "Raw Reflection",
  "currentSizeBytes": 0,
  "outputRecords": -1,
  "totalSizeBytes": 0,
  "datasetId": "acd4d4be-7049-47e4-b616-b471c5b3c60c",
  "datasetType": "PHYSICAL_DATASET",
  "datasetPath": [
    "@dremio",
    "test"
  ]
}

```

```

    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "GIVEN_UP",
      "availabilityStatus": "NONE",
      "combinedStatus": "FAILED",
      "refreshMethod": "NONE",
      "failureCount": 3,
      "lastDataFetchAt": null,
      "expiresAt": null,
      "lastRefreshDurationMillis": -1
    },
    "consideredCount": 0,
    "matchedCount": 0,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22d7cc7745fad2-be18-40a4-1922-a567d3e0%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:19.030Z",
    "updatedAt": "2023-01-13T19:50:19.030Z",
    "id": "8eec62d7-3419-4cf3-997d-0a153d81ed8a",
    "reflectionType": "AGGREGATION",
    "name": "dataset991_agg991",
    "currentSizeBytes": 9273,
    "outputRecords": 51,
    "totalSizeBytes": 9273,
    "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
    "datasetType": "VIRTUAL_DATASET",
    "datasetPath": [
      "temp",
      "dataset991"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "MANUAL",
      "availabilityStatus": "AVAILABLE",
      "combinedStatus": "CAN_ACCELERATE",
      "refreshMethod": "FULL",
      "failureCount": 0,
      "lastDataFetchAt": "2023-01-13T19:46:02.342Z",
      "expiresAt": "3022-05-16T19:46:02.342Z",
      "lastRefreshDurationMillis": 11697
    }
  }

```

```

    },
    "consideredCount": 60,
    "matchedCount": 9,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":
"/jobs?filters=%7B%22chr%22%3A%5B%22a8de18d351a0-d799-3fc4-9143-7d26cee8%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "isArrowCachingEnabled": false,
    "isCanView": true,
    "isCanAlter": true,
    "isEnabled": true
  },
  {
    "createdAt": "2023-01-13T19:50:17.714Z",
    "updatedAt": "2023-01-13T19:50:17.714Z",
    "id": "167428eb-7936-4ea2-a1fb-23b1ac6e9454",
    "reflectionType": "RAW",
    "name": "dataset991_raw991",
    "currentSizeBytes": 818790,
    "outputRecords": 29467,
    "totalSizeBytes": 818790,
    "datasetId": "a461bf97-8464-43ed-bd86-a8fb90d920e3",
    "datasetType": "VIRTUAL_DATASET",
    "datasetPath": [
      "temp",
      "dataset991"
    ],
    "status": {
      "configStatus": "OK",
      "refreshStatus": "MANUAL",
      "availabilityStatus": "AVAILABLE",
      "combinedStatus": "CAN_ACCELERATE",
      "refreshMethod": "FULL",
      "failureCount": 0,
      "lastDataFetchAt": "2023-01-13T19:46:02.747Z",
      "expiresAt": "3022-05-16T19:46:02.747Z",
      "lastRefreshDurationMillis": 16666
    },
    "consideredCount": 54,
    "matchedCount": 37,
    "chosenCount": 0,
    "consideredJobsLink":
"/jobs?filters=%7B%22cor%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "matchedJobsLink":
"/jobs?filters=%7B%22mar%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%2
2%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
    "chosenJobsLink":

```



```
"/jobs?filters=%7B%22chr%22%3A%5B%224549e6ca1b32-bf1a-2ae4-6397-be824761%22%5D%2C%22qt%22%3A%5B%22UI%22%2C%22EXTERNAL%22%2C%22ACCELERATION%22%5D%7D",
  "isArrowCachingEnabled": false,
  "isCanView": true,
  "isCanAlter": true,
  "isEnabled": true
}
],
"nextPageToken": "CiQxNjc0Mjhlyi030TM2LTRLyTItYTfmYi0yM2IxYWM2ZTk0NTQSAA==",
"isCanAlterReflections": true
}
```

This endpoint supports query parameters that you can add to the request URL to include only specific types of Reflections in the Reflection summary, specify the maximum number of results to return, and sort the response to list Reflections in ascending or descending order.

For example, to order the Reflections within the summary in ascending order by reflectionName, add `?orderBy=reflectionName` to the request URL. For descending order, add a `-` character before the attribute name: `?orderBy=-reflectionName`.

In the same request, you can add the `filter` query parameter to retrieve only the raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths. The JSON object for such a filter would look like this:

Example JSON Object for Filter

```
{
  "reflectionType": ["RAW"],
  "refreshStatus": ["MANUAL", "SCHEDULED"],
  "enabledFlag": true,
  "reflectionNameOrDatasetPath": "samples.dremio.com"
}
```

However, to use the JSON object in the request URL, you must convert it to URL-encoded JSON, which looks like this:

Example JSON Object in URL-Encoded JSON

```
%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflectionNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D
```

Here is an example request URL that includes both the `orderBy` and `filter` query parameters:

Example Request with orderBy and filter Query Parameters

```
curl -X GET
'https://{hostname}/api/v3/reflection-summary?orderBy=reflectionName&filter=%7B%0A%20%20%22reflectionType%22%3A%20%5B%22RAW%22%5D%2C%0A%20%20%22refreshStatus%22%3A%20%5B%22MANUAL%22%2C%22SCHEDULED%22%5D%2C%0A%20%20%22enabledFlag%22%3A%20true%2C%0A%20%20%22reflecti
```

```
onNameOrDatasetPath%22%3A%20%22samples.dremio.com%22%0A%7D' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

For this request, the Reflection summary in the response will include only raw Reflections that are refreshed manually or by schedule, are enabled, and apply to datasets with `samples.dremio.com` in their paths, and the Reflections will be listed in ascending order by reflectionName.

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Recommendations](/current/reference/api/reflections/reflection-recommendations)

Reflection Summary Attributes

Retrieve a Reflection Summary

Parameters

Example Response

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/roles/privilege>

Version: current [26.x]

On this page

Role Privileges Enterprise

Use the Role API to retrieve information about the privileges assigned to roles.

Role Privileges Object

```
{
  "data": [
    {
      "name": "hive",
```

```

    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "CREATE_TABLE"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "INSERT"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "DROP"
  },
  {
    "name": "hive",
    "type": "SOURCE",
    "privilege": "ALTER"
  },
  {
    "name": "mysql",
    "type": "SOURCE",
    "privilege": "EXTERNAL_QUERY"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "SELECT"
  },
  {
    "name": "Samples.\"samples.dremio.com\"",
    "type": "FOLDER",
    "privilege": "SELECT"
  },
  {
    "name": "Samples.\"samples.dremio.com\".\"SF_incidents2016.json\"",
    "type": "DATASET",
    "privilege": "SELECT"
  },
  {
    "name": "testing",
    "type": "SPACE",
    "privilege": "SELECT"
  }
]
}

```

Role Privileges Attributes

data Array of Object

Information about the privileges the specified role has for the entities in the current organization, up to a maximum of 100 privileges. Each object in the data array describes a privilege the role has for a specific entity.

Attributes of Objects in the `data` Array

name String (UUID)

Name of the object to which the privilege applies. The name includes the objects's parent objects, if any.

Example: hive

type String

Type of the object to which the privilege applies.

Enum: SPACE, SOURCE, HOME, FOLDER, DATASET, FUNCTION

Example: SOURCE

privilege String

Name of the privilege that the role has for the object. Available privileges vary for different object types.

Example: SELECT

Retrieve Role Privileges

Retrieve the specified role's privileges.

Method and URL

```
GET /api/v3/role/{id}/privilege
```

Parameters

id Path String (UUID)

Unique identifier of the role whose privileges you want to retrieve.

Example: 3d83e7d7-98ee-4afa-ebdd-41c30eb92744

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/role/3d83e7d7-98ee-4afa-ebdd-41c30eb92744/privilege' \
--header 'Authorization: Bearer <dremioAccessToken>' \
```

```
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "name": "hive",
      "type": "SOURCE",
      "privilege": "SELECT"
    },
    {
      "name": "hive",
      "type": "SOURCE",
      "privilege": "CREATE_TABLE"
    },
    {
      "name": "hive",
      "type": "SOURCE",
      "privilege": "INSERT"
    },
    {
      "name": "hive",
      "type": "SOURCE",
      "privilege": "DROP"
    },
    {
      "name": "hive",
      "type": "SOURCE",
      "privilege": "ALTER"
    },
    {
      "name": "mysql",
      "type": "SOURCE",
      "privilege": "EXTERNAL_QUERY"
    },
    {
      "name": "Samples",
      "type": "SOURCE",
      "privilege": "SELECT"
    },
    {
      "name": "Samples.\"samples.dremio.com\"",
      "type": "FOLDER",
      "privilege": "SELECT"
    },
    {
      "name": "Samples.\"samples.dremio.com\".\"SF_incidents2016.json\"",
      "type": "DATASET",
      "privilege": "SELECT"
    },
    {
      "name": "testing",
```

```
    "type": "SPACE",  
    "privilege": "SELECT"  
  }  
]  
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Role](/current/reference/api/roles/)

Role Privileges Attributes

Retrieve Role Privileges

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/user/credentials>

Version: current [26.x]

On this page

Credentials Enterprise

Use the Credentials API to manage the credentials of Dremio service users. Dremio offers two forms of service user credentials:

Client Credentials – A Client ID and Client Secret assigned by Dremio.

External Credentials – The definition of external JSON Web Tokens (JWTs) from an integrated enterprise identity provider.

Client Credentials

External Credentials

Client Credential Object

```
{
  "id": "88ec35b1-672c-4087-9e21-5b6d8bff4044",
  "name": "testing-credential",
  "clientSecretConfig": {
    "clientId": "61d00e34-a938-4b80-9a98-063fc723bd6d",
    "clientSecret": "wSWwbLG...6akQ",
    "expiresAt": "2026-01-06T16:03:39.527Z",
    "createdAt": "2025-10-08T16:03:39.527Z"
  },
  "credentialType": "CLIENT_SECRET"
}
```

External Credential Object

```
{
  "id": "3c3a13dd-d0b7-4dd8-b76c-4633e1f8635e",
  "name": "new-external-credential",
  "externalJwtCredentialConfig": {
    "issuer": "https://sts.windows.net/3e334762-b0c6-4c36-9faf-93800f0d6c71/",
    "allowedAudiences": [
      "api://b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
    ],
    "identifierClaim": "oid",
    "identifierClaimValue": "ad5be351-6397-439e-ad83-2f19e0777ecf",
    "tokenExchangeAudience":
      "https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/discovery/keys?appid=b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
  },
  "credentialType": "EXTERNAL_JWT"
}
```

Attributes

id String (UUID)

Unique identifier of the credential.

name String

Name of the credential. This name is not required to be unique.

credentialType String

Type of the credential. Valid values are `CLIENT_SECRET` and `EXTERNAL_JWT`.

clientSecretConfig Object

The configuration attributes of the client secret.

externalJwtCredentialConfig Object

The configuration attributes of the external JWT when the service user is authenticated by an external identity provider.

Attributes of the `clientSecretConfig` Object

clientId String (UUID)

The client ID assigned to the service user of this credential. The client ID is assigned at the time of service user creation and is immutable.

clientSecret String

The client secret of this credential. The client secret is included in the return object from a create operation and is never displayed again.

expiresAt String

Timestamp of the expiration of the credential. The lifetime of the credential is configured in the creation request.

createdAt String

Timestamp of the creation of the credential.

Attributes of the `externalJwtCredentialConfig` Object

issuer String

A URI that describes who issued the token. For tokens from Microsoft Entra ID, the issuer will be a Microsoft endpoint.

Example: `https://sts.windows.net/<tenant-id>`

allowedAudiences Array of String

List of URIs describing the intended audience for the token. For tokens from Microsoft Entra ID, this field is the Application ID URI. The default format is `api://<client-id>`.

identifierClaim String

A mapping that describes the field in the token containing the service user identity. For tokens from Microsoft Entra ID, the appropriate fields are `sub` or `oid`. See [Microsoft documentation](#) for additional information.

Example: `oid`

identifierClaimValue String

The user identifier from the external identity provider.

tokenExchangeAudience String

A URI that identifies the target resource for token exchange and where the exchanged token is intended to be used. The URI includes the client ID of the service user and the

credential ID.

jwksUri String

The URI from which to retrieve the JWKS, used to validate the signature of the token. If none is provided, Dremio will default to using ``<issuer_URL>/well-known/openid-configuration``. For tokens from Microsoft Entra ID, the JWKS URI is not found under the issuer, so it must be provided. The default location URI is ``https://login.microsoftonline.com/<tenant-id>/discovery/keys``.

Example: ``https://login.microsoftonline.com/<tenant-id>/discovery/keys``

List All Credentials

Method and URL

```
GET /api/v3/user/{id}/oauth/credentials
```

Parameters

id Path String (UUID)

Unique identifier of the service user.

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/user/5263eb65-5fba-406b-9539-8627240adb8e/oauth/credentials' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "88ec35b1-672c-4087-9e21-5b6d8bffa044",
  "name": "testing-credential",
  "clientSecretConfig": {
    "clientId": "61d00e34-a938-4b80-9a98-063fc723bd6d",
    "expiresAt": "2026-01-06T16:03:39.527Z",
    "createdAt": "2025-10-08T16:03:39.527Z"
  },
  "credentialType": "CLIENT_SECRET"
}
```

Response Status Codes

200 OK
400 Bad Request
401 Unauthorized
404 Not Found
500 Internal Server Error

Create a Client Credential

Method and URL

POST /api/v3/user/{id}/oauth/credentials

Parameters

id Path String (UUID)

Unique identifier of the service user.

credentialType Body String

The type of credential to create. Must be `CLIENT_SECRET` for a client credential.

name Body String

The name of the credential.

Example: `new-credential`

clientSecretConfig Body Object

Configuration of the client secret to be created. Contains the credential lifetime, specified as quantity (an integer with a maximum value of 180) and units (must be `DAYS`).

Example

Request

```
curl -X POST \
'https://{hostname}/api/v3/user/5263eb65-5fba-406b-9539-8627240adb8e/oauth/credentials' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "credentialType": "CLIENT_SECRET",
  "name": "new-credential",
  "clientSecretConfig": {
    "expiresIn": {
      "quantity": 90,
      "units": "DAYS"
    }
  }
}
```

```
}  
}  
'
```

Response

```
{  
  "id": "c125b06c-b187-439b-8f71-df8cc4784835",  
  "name": "new-credential",  
  "clientSecretConfig": {  
    "clientId": "61d00e34-a938-4b80-9a98-063fc723bd6d",  
    "clientSecret": "wSWwbLG...6akQ",  
    "expiresAt": "2026-01-13T07:47:11.490Z",  
    "createdAt": "2025-10-15T07:47:11.490Z"  
  },  
  "credentialType": "CLIENT_SECRET"  
}
```

Response Status Codes

201 Created

400 Bad Request

401 Unauthorized

404 Not Found

500 Internal Server Error

Create an External Credential

Method and URL

```
POST /api/v3/user/{id}/oauth/credentials
```

Parameters

id Path String (UUID)

Unique identifier of the service user.

credentialType Body String

The type of credential to create. Must be `EXTERNAL_JWT` for an external credential.

name Body String

The name of the credential.

Example: `new-external-credential`

externalJwtCredentialConfig Body Object

The configuration of the external JWT.

Parameters of the `externalJwtCredentialConfig` Object

issuer String

A URI that describes who issued the token. For tokens from Microsoft Entra ID, the issuer will be a Microsoft endpoint.

Example: `https://sts.windows.net/<tenant-id>`

allowedAudiences Array of String

List of URIs describing the intended audience for the token. For tokens from Microsoft Entra ID, this field is the Application ID URI. The default format is `api://<client-id>`.

identifierClaim String

A mapping that describes the field in the token containing the service user identity. For tokens from Microsoft Entra ID, the appropriate fields are `sub` or `oid`. See [Microsoft documentation](#) for additional information.

Example: `oid`

identifierClaimValue String

The user identifier from the external identity provider.

jwtUri String

The URI from which to retrieve the JWKS, used to validate the signature of the token. If none is provided, Dremio will default to using `

Example: `https://login.microsoftonline.com/<tenant-id>/discovery/keys`

Example

Request

```
curl -X POST \
'https://{hostname}/api/v3/user/5263eb65-5fba-406b-9539-8627240adb8e/oauth/credentials' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "credentialType": "EXTERNAL_JWT",
  "name": "new-external-credential",
  "externalJwtCredentialConfig": {
    "issuer": "https://sts.windows.net/3e334762-b0c6-4c36-9faf-93800f0d6c71/",
```

```

    "allowedAudiences": [
      "api://b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
    ],
    "identifierClaim": "oid",
    "identifierClaimValue": "ad5be351-6397-439e-ad83-2f19e0777ecf",
                                                                    "jwksUri":
"https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/discovery/keys?appid=b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
  }
}'

```

Response

```

{
  "id": "3c3a13dd-d0b7-4dd8-b76c-4633e1f8635e",
  "name": "new-external-credential",
  "externalJwtCredentialConfig": {
    "issuer": "https://sts.windows.net/3e334762-b0c6-4c36-9faf-93800f0d6c71/",
    "allowedAudiences": [
      "api://b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
    ],
    "identifierClaim": "oid",
    "identifierClaimValue": "ad5be351-6397-439e-ad83-2f19e0777ecf",
                                                                    "tokenExchangeAudience":
"//oauth.dremio.app/clients/61d00e34-a938-4b80-9a98-063fc723bd6d/credentials/3c3a13dd-d0b7-4dd8-b76c-4633e1f8635e",
                                                                    "jwksUri":
"https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/discovery/keys?appid=b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
  },
  "credentialType": "EXTERNAL_JWT"
}

```

Response Status Codes

- 201 Created
- 400 Bad Request
- 401 Unauthorized
- 404 Not Found
- 500 Internal Server Error

Update an External Credential

Method and URL

```
PUT /api/v3/user/{id}/oauth/credentials/{credentialId}
```

Parameters

id Path String (UUID)

Unique identifier of the service user.

credentialId Path String (UUID)

Unique identifier of the credential.

credentialType Body String

The type of credential to update. Must be `EXTERNAL_JWT` for an external credential.

name Body String

The name of the credential. The name may be updated in this operation.

Example: `updated-external-credential`

externalJwtCredentialConfig Body Object

The configuration of the external JWT. Any field can be updated by this operation.

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/user/5263eb65-5fba-406b-9539-8627240adb8e/oauth/credentials/3c3a13dd-d0b7-4dd8-b76c-4633e1f8635e' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "credentialType": "EXTERNAL_JWT",
  "name": "updated-external-credential",
  "externalJwtCredentialConfig": {
    "issuer": "https://sts.windows.net/3e334762-b0c6-4c36-9faf-93800f0d6c71/",
    "allowedAudiences": [
      "api://b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
    ],
    "identifierClaim": "oid",
    "identifierClaimValue": "ad5be351-6397-439e-ad83-2f19e0777ecf",
    "jwksUri":
      "https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/discovery/keys?appid=b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
  }
}'
```

Response

```
{
  "id": "3c3a13dd-d0b7-4dd8-b76c-4633e1f8635e",
  "name": "updated-external-credential",
  "externalJwtCredentialConfig": {
```

```

"issuer": "https://sts.windows.net/3e334762-b0c6-4c36-9faf-93800f0d6c71/",
"allowedAudiences": [
  "api://b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
],
"identifierClaim": "oid",
"identifierClaimValue": "ad5be351-6397-439e-ad83-2f19e0777ecf",
"tokenExchangeAudience":
  "https://login.microsoftonline.com/3e334762-b0c6-4c36-9faf-93800f0d6c71/discovery/keys?appid=b0f6e5d4-6e6b-40d9-a9c2-33bb78ceb880"
},
"credentialType": "EXTERNAL_JWT"
}

```

Response Status Codes

- 200 OK
- 400 Bad Request
- 401 Unauthorized
- 404 Not Found
- 500 Internal Server Error

Delete a Credential

Method and URL

```
DELETE /api/v3/user/{id}/oauth/credentials/{credentialId}
```

Parameters

id Path String (UUID)

Unique identifier of the service user.

credentialId Path String (UUID)

Unique identifier of the credential. This endpoint accepts both client credentials and external credentials.

Example

Request

```
curl -X DELETE
```

```
'https://{hostname}/api/v3/user/5263eb65-5fba-406b-9539-8627240adb8e/oauth/credentials/c125b06c-b187-439b-8f71-df8cc4784835' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

A successful request returns an empty response with the HTTP `204 No Content` status.

Response Status Codes

204 No Content

400 Bad Request

403 Forbidden

500 Internal Server Error

Was this page helpful?

[Previous

User]/(/current/reference/api/user/)[Next

User Privileges]/(/current/reference/api/user/privilege)

Attributes

List All Credentials

Parameters

Example

Response Status Codes

Create a Client Credential

Parameters

Example

Response Status Codes

Create an External Credential

Parameters

Example

Response Status Codes

Update an External Credential

Parameters

-

Example

Response Status Codes

Delete a Credential

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/user/privilege>

Version: current [26.x]

On this page

User Privileges Enterprise

Use the User API to retrieve privilege information for Dremio users.

User Privileges Object

```
{
  "data": [
    {
      "name": "\"@dremio\".\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\"",
      "type": "PDS",
      "privilege": "ALTER"
    },
    {
      "name": "\"@dremio\".\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\"",
      "type": "PDS",
      "privilege": "SELECT"
    },
    {
      "name": "\"@dremio\".Business",
      "type": "FOLDER",
      "privilege": "ALTER"
    },
    {
      "name": "\"@dremio\".Business",
      "type": "FOLDER",
      "privilege": "SELECT"
    },
    {
      "name": "Samples",
      "type": "SOURCE",
      "privilege": "SELECT"
    },
    {

```

```

    "name": "Samples",
    "type": "SOURCE",
    "privilege": "INSERT"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "DROP"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "CREATE_TABLE"
  },
  {
    "name": "Samples",
    "type": "SOURCE",
    "privilege": "ALTER"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "SELECT"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "MANAGE_GRANTS"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "MODIFY"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "ALTER_REFLECTION"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "VIEW_REFLECTION"
  },
  {
    "name": "\"testing\"",
    "type": "SPACE",
    "privilege": "ALTER"
  },
  {
    "name": "\"testing\".\"NYC-taxi-trips\"",
    "type": "VDS",
    "privilege": "SELECT"
  },

```

```

{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "ALTER"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "DELETE"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "MODIFY"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "VIEW"
}
]
}

```

User Privileges Attributes

data Array of Object

Information about the privileges the specified user has for the catalog objects in the current organization, up to a maximum of 100 privileges. Each object in the data array describes a privilege on a specific catalog object.

Attributes of Objects in the `data` Array

name String (UUID)

Name of the object to which the privilege applies. The name includes the entity's parent space or folder, if any.

Example: "@dremio"."1c0accd3-e8c0-1d55-23a2-0ff6529f6c00"

type String

Type of the object to which the privilege applies.

Enum: SPACE, SOURCE, HOME, FOLDER, PDS, VDS, FUNCTION

Example: PDS

privilege String

Name of the privilege that the user has for the object. Available privileges vary for different object types.

Example: ALTER

Retrieve User Privileges

Retrieve the specified user's privileges.

Method and URL

```
GET /api/v3/user/{id}/privilege
```

Parameters

id Path String

Unique identifier of the user whose privileges you want to retrieve.

Example: b9dbebc7-bc3b-4d56-9154-31762ab65a43

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/user/b9dbebc7-bc3b-4d56-9154-31762ab65a43/privilege' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "name": "\"@dremio\".\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\"",
      "type": "PDS",
      "privilege": "ALTER"
    },
    {
      "name": "\"@dremio\".\"1c0accd3-e8c0-1d55-23a2-0ff6529f6c00\"",
      "type": "PDS",
      "privilege": "SELECT"
    }
  ],
}
```

```

{
  "name": "\"@dremio\".Business",
  "type": "FOLDER",
  "privilege": "ALTER"
},
{
  "name": "\"@dremio\".Business",
  "type": "FOLDER",
  "privilege": "SELECT"
},
{
  "name": "Samples",
  "type": "SOURCE",
  "privilege": "SELECT"
},
{
  "name": "Samples",
  "type": "SOURCE",
  "privilege": "INSERT"
},
{
  "name": "Samples",
  "type": "SOURCE",
  "privilege": "DROP"
},
{
  "name": "Samples",
  "type": "SOURCE",
  "privilege": "CREATE_TABLE"
},
{
  "name": "Samples",
  "type": "SOURCE",
  "privilege": "ALTER"
},
{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "SELECT"
},
{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "MODIFY"
},
{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "ALTER_REFLECTION"
}

```

```

},
{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "VIEW_REFLECTION"
},
{
  "name": "\"testing\"",
  "type": "SPACE",
  "privilege": "ALTER"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "SELECT"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "\"testing\".\"NYC-taxi-trips\"",
  "type": "VDS",
  "privilege": "ALTER"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "MANAGE_GRANTS"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "DELETE"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "MODIFY"
},
{
  "name": "taxi",
  "type": "SCRIPT",
  "privilege": "VIEW"
}
]
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Credentials][/current/reference/api/user/credentials)[Next

Workload Management][current/reference/api/wlm/)

User Privileges Attributes

Retrieve User Privileges

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/wlm/queue>

Version: current [26.x]

On this page

Queue Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM queues.

Queue Object

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",
      "name": "High Cost Reflections",
      "maxMemoryPerNodeBytes": 8589934592,
      "maxQueryMemoryPerNodeBytes": 8589934592,
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 300000,
      "maxRunTimeoutMs": 300000,
      "engineId": "DATA"
    },
    {
      "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "tag": "HM2D9XElG3U=",
```

```

    "name": "Low Cost Reflections",
    "cpuTier": "BACKGROUND",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 86400000
  },
  {
    "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "tag": "p22KaFcaB7g=",
    "name": "COPY & OPTIMIZATION Queue",
    "maxMemoryPerNodeBytes": 4294967296,
    "maxQueryMemoryPerNodeBytes": 4294967296,
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 2,
    "maxStartTimeoutMs": 300000,
    "engineId": "YARN"
  },
  {
    "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "tag": "//gNL3Ta2bY=",
    "name": "Low Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 100,
    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Queue Attributes

data Array of Object

List of queue objects in the Dremio instance.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the queue, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag String

Unique identifier of the version of the queue. Dremio changes the tag whenever the

queue changes and uses the tag to ensure that PUT requests apply to the most recent version of the queue.

Example: BNGRmgfEnDg=

name String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Integer

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

maxQueryMemoryPerNodeBytes Integer

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier String

Amount of CPU time that threads get compared to other threads.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Integer

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Integer

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Integer

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId String

Name of the execution engine to which the queue's queries are routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. The engineId attribute is omitted from the queue object if no engine is specified.

Example: DATA

Create a Queue

Create a WLM queue.

Method and URL

POST /api/v3/wlm/queue

Parameters

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId Body String Optional

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution.

Example: DATA

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "High Cost Reflections",
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "maxMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Retrieve All Queues

Retrieve all WLM queues.

Method and URL

```
GET /api/v3/wlm/queue
```

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",
      "name": "High Cost Reflections",
      "maxMemoryPerNodeBytes": 8589934592,
      "maxQueryMemoryPerNodeBytes": 8589934592,
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 300000,
      "maxRunTimeoutMs": 300000,
      "engineId": "DATA"
    },
    {
      "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "tag": "HM2D9XElG3U=",
      "name": "Low Cost Reflections",
      "cpuTier": "BACKGROUND",
      "maxAllowedRunningJobs": 10,
      "maxStartTimeoutMs": 86400000
    },
    {
      "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "tag": "p22KaFcaB7g=",
      "name": "COPY & OPTIMIZATION Queue",
      "maxMemoryPerNodeBytes": 4294967296,
      "maxQueryMemoryPerNodeBytes": 4294967296,
      "cpuTier": "MEDIUM",
      "maxAllowedRunningJobs": 2,
      "maxStartTimeoutMs": 300000,
    }
  ]
}
```

```

    "engineId": "YARN"
  },
  {
    "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "tag": "//gNL3Ta2bY=",
    "name": "Low Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 100,
    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a Queue by ID

Retrieve a specific WLM queue by the queue's ID.

Method and URL

```
GET /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to retrieve, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a Queue by Name

Retrieve a specific WLM queue by the queue's name.

Method and URL

```
GET /api/v3/wlm/queue/by-name/{name}
```

Parameters

name Path String

Name for the queue you want to retrieve. If the queue name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: High%20Cost%20Reflections

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/by-name/High%20Cost%20Reflections' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Update a Queue

Update the specified WLM queue.

Method and URL

```
PUT /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to update, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag Body String

Unique identifier of the version of the queue to update. Dremio uses the tag to ensure that you are updating the most recent version of the queue.

Example: BNGRmgfEnDg=

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node. If you omit the maxMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node. If you omit the maxQueryMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxQueryMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxQueryMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`. If you omit the cpuTier parameter in a PUT request, Dremio replaces it with the default value. To keep the existing setting while making other updates, include the existing cpuTier parameter and setting in the PUT request.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: LOW

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel. If you omit the maxAllowedRunningJobs parameter in a PUT request, Dremio removes the existing maxAllowedRunningJobs value from the queue. To keep the existing value while making other updates, include the existing maxAllowedRunningJobs parameter and value in the PUT request.

Example: 100

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds. If you omit the maxStartTimeoutMs parameter in a PUT request, Dremio removes the existing maxStartTimeoutMs setting from the queue. To keep the existing setting while making other updates, include the existing maxStartTimeoutMs parameter and setting in the PUT request.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query is allowed to run before it is cancelled, in milliseconds. If you omit the maxRunTimeoutMs parameter in a PUT request, Dremio removes the existing maxRunTimeoutMs value from the queue. To keep the existing value while making other updates, include the existing maxRunTimeoutMs parameter and value in the PUT request.

Example: 300000

engineId Body String

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution.

Example: DATA

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "LOW",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "xQh6KNyEjus=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
```

```
"maxQueryMemoryPerNodeBytes": 8589934592,  
"cpuTier": "LOW",  
"maxAllowedRunningJobs": 100,  
"maxStartTimeoutMs": 300000,  
"maxRunTimeoutMs": 300000,  
"engineId": "DATA"  
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

500 Internal Server Error

Delete a Queue

Delete the specified WLM queue.

Method and URL

```
DELETE /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue that you want to delete, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example

Request

```
curl -X DELETE  
'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Workload Management](/current/reference/api/wlm/)[Next

Rule](/current/reference/api/wlm/rule)

Queue Attributes

Create a Queue

Parameters

Example

Response Status Codes

Retrieve All Queues

Example

Response Status Codes

Retrieve a Queue by ID

Parameters

Example

Response Status Codes

Retrieve a Queue by Name

Parameters

Example

Response Status Codes

Update a Queue

Parameters

Example

Response Status Codes

Delete a Queue

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/wlm/rule>

Version: current [26.x]

On this page

Rule Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM rules.

The rule object includes a rules array (also called the ruleset). Each object in the rules array represents an individual rule. Dremio processes rules in the order they are listed within the rules array: the highest-priority rule is listed first, and the lowest-priority rule is listed last.

Rule Object

```
{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    }
  ],
}
```

```

{
  "name": "COPY & OPTIMIZATION Rule",
  "conditions": "query_label() in ('COPY','OPTIMIZATION')",
  "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
  "acceptName": "COPY & OPTIMIZATION Queue",
  "action": "PLACE",
  "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
},
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 300000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Rule Attributes

tag String

Unique identifier of the version of the rule. Dremio changes the tag whenever a rule changes and uses the tag to ensure that PUT requests apply to the most recent version of the rules.

Example: VmqwaZ90VY4=

rules Array of Object

List of rule objects in the Dremio instance.

defaultRule Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule.

Example: {"name": "All Other Queries","action": "REJECT","id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}

Attributes of Objects in the `rules` Array

name String

User-provided name for the rule.

Example: High Cost Reflections

conditions String

Conditions that queries must match to be placed in the queue.

Example: `query_type() = 'Reflections' AND query_cost() >= 30000000`

acceptId String (UUID)

Unique identifier of the queue in which the rule places queries.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

acceptName Integer

User-provided name for the queue in which the rule places queries.

Example: High Cost Reflections

rejectMessage Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Integer

Action the rule takes for queries that match the rule conditions.

Enum: PLACE, REJECT

Example: PLACE

id String (UUID)

Unique identifier of the rule, in UTC format.

Example: fa1ec87d-923b-414c-9064-e079f39f5c49

Attributes of the `defaultRule` Object

name String

User-provided name for the default rule.

Example: All Other Queries

acceptId String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which

the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action String

Action the default rule takes for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

Create or Update a Rule

Create or update a WLM rule.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To add or update an individual rule, you must include the entire rules array in the request body. It is not necessary to specify the ID for the rule you want to delete in the request URL.

Method and URL

```
PUT /api/v3/wlm/rule
```

Parameters

tag Body String

Unique identifier of the rules instance. Dremio uses the tag to ensure that PUT requests apply to the most recent version of the rules. Omit if you are creating rules for the organization for the first time.

Example: VmqwaZ90VY4=

rules Body Array of Object

List of rule objects in the Dremio instance. To add or update an individual rule, you must include the entire rules array in the request body.

Example:

defaultRule Body Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule. To add or update an individual rule, you must include the defaultRule object in the request body.

Example: {"name": "All Other Queries","action": "REJECT","id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}

Parameters of Objects in the `rules` Array

name Body String

User-provided name for the rule.

Example: DevOps and Engineering

conditions Body String

Conditions that queries must match to be placed in the queue.

Example: is_member('DevOps') OR is_member('Engineering')

acceptId Body String (UUID)

For rules whose action is `PLACE`, the unique identifier of the queue in which the rule should place queries.

Example: b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1

acceptName Body Integer Optional

For rules whose action is `PLACE`, the user-provided name for the queue in which the rule should place queries.

Example: High Cost Reflections

rejectMessage Body Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Body Integer Optional

Action the rule should take for queries that match the rule conditions. Default is `PLACE`.

Enum: PLACE, REJECT

Example: PLACE

Parameters of the `defaultRule` Object

name Body String

User-provided name for the default rule.

Example: All Other Queries

acceptId Body String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Body Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Body Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action Body String

Action the default rule should take for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id Body String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

This example request demonstrates how to add a `DevOps and Engineering` rule to the ruleset:

Example Request to Add a Rule

```
curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
```

```

    "name": "DevOps and Engineering",
    "conditions": "is_member('DevOps') OR is_member('Engineering')",
    "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
    "acceptName": "DevOps and Eng Testing",
    "action": "PLACE"
  },
  {
    "name": "High Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
    "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
    "acceptName": "High Cost Reflections",
    "action": "PLACE",
    "id": "falec87d-923b-414c-9064-e079f39f5c49"
  },
  {
    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY', 'OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 30000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 30000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'

```

Example Response

```
{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE",
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ]
},
```

```

"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

To change the order in which Dremio processes rules, send a PUT request that lists the rules in the desired order within the rules array, with the highest-priority rule listed first, and the lowest-priority rule listed last.

This example request reorders the rules so that the `COPY & OPTIMIZATION Rule` will be the highest-priority rule:

Example Request to Reorder Rules

```

curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
  ],
}

```

```
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
},
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'
```

Example Response

```
{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 300000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    }
  ]
}
```

```

    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 300000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ],
  "defaultRule": {
    "name": "All Other Queries",
    "action": "REJECT",
    "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
  }
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve All Rules

Retrieve all WLM rules.

Method and URL

```
GET /api/v3/wlm/rule
```

Example

Request

```

curl -X GET 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'

```

Response

```
{
```

```

"tag": "3uzixTFD134=",
"rules": [
  {
    "name": "DevOps and Engineering",
    "conditions": "is_member('DevOps') OR is_member('Engineering')",
    "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
    "acceptName": "DevOps and Eng Testing",
    "action": "PLACE",
    "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"
  },
  {
    "name": "High Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
    "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
    "acceptName": "High Cost Reflections",
    "action": "PLACE",
    "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
  },
  {
    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 30000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 30000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",

```

```
"id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}
```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Delete a Rule

Delete a WLM rule from the ruleset.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To delete a rule, send a PUT request that omits the rule from the rules array. It is not necessary to specify the ID for the rule you want to delete in the request URL.

note

The default rule can be updated but cannot be deleted.

Method and URL

```
PUT /api/v3/wlm/rule/
```

This example request demonstrates how to remove the `DevOps and Engineering` rule added in the Creating or Updating a Rule example:

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
```



```

    "action": "PLACE",
    "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
  },
  {
    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'

```

Response

```

{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",

```

```

    "action": "PLACE",
    "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
  },
  {
    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 30000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 30000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Was this page helpful?

[Previous

Queue](/current/reference/api/wlm/queue)

Rule Attributes

Create or Update a Rule

Parameters

Example Response

Example Response

Response Status Codes

Retrieve All Rules

Example

Response Status Codes

Delete a Rule

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/wlm/queue/>

Version: current [26.x]

On this page

Queue Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM queues.

Queue Object

```
{
  "data": [
    {
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "tag": "BNGRmgfEnDg=",
      "name": "High Cost Reflections",
      "maxMemoryPerNodeBytes": 8589934592,
      "maxQueryMemoryPerNodeBytes": 8589934592,
      "cpuTier": "BACKGROUND",
```

```

    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000,
    "maxRunTimeoutMs": 300000,
    "engineId": "DATA"
  },
  {
    "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "tag": "HM2D9XElG3U=",
    "name": "Low Cost Reflections",
    "cpuTier": "BACKGROUND",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 86400000
  },
  {
    "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "tag": "p22KaFcaB7g=",
    "name": "COPY & OPTIMIZATION Queue",
    "maxMemoryPerNodeBytes": 4294967296,
    "maxQueryMemoryPerNodeBytes": 4294967296,
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 2,
    "maxStartTimeoutMs": 300000,
    "engineId": "YARN"
  },
  {
    "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "tag": "//gNL3Ta2bY=",
    "name": "Low Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 100,
    "maxStartTimeoutMs": 300000
  },
  {
    "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "tag": "wa+vYmA73gU=",
    "name": "High Cost User Queries",
    "cpuTier": "MEDIUM",
    "maxAllowedRunningJobs": 10,
    "maxStartTimeoutMs": 300000
  }
]
}

```

Queue Attributes

data Array of Object

List of queue objects in the Dremio instance.

Attributes of Objects in the `data` Array

id String (UUID)

Unique identifier of the queue, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag String

Unique identifier of the version of the queue. Dremio changes the tag whenever the queue changes and uses the tag to ensure that PUT requests apply to the most recent version of the queue.

Example: BNGRmgfEnDg=

name String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Integer

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

maxQueryMemoryPerNodeBytes Integer

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier String

Amount of CPU time that threads get compared to other threads.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Integer

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Integer

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Integer

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId String

Name of the execution engine to which the queue's queries are routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution. The engineId attribute is omitted from the queue object if no engine is specified.

Example: DATA

Create a Queue

Create a WLM queue.

Method and URL

```
POST /api/v3/wlm/queue
```

Parameters

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: BACKGROUND

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel.

Example: 10

maxStartTimeoutMs Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds.

Example: 300000

maxRunTimeoutMs Body Integer Optional

Maximum length of time that a query can run before it is cancelled, in milliseconds.

Example: 300000

engineId Body String Optional

Name of the execution engine to which the queue's queries should be routed. If you do not specify an engineId, the queue's queries run on any engine that is available at the time of execution.

Example: DATA

Example

Request

```
curl -X POST 'https://{hostname}/api/v3/wlm/queue' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "High Cost Reflections",
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "maxMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request
401 Unauthorized
403 Forbidden
404 Not Found
405 Method Not Allowed
409 Conflict

Retrieve All Queues

Retrieve all WLM queues.

Method and URL

```
GET /api/v3/wlm/queue
```

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
{  
  "data": [  
    {  
      "id": "1990e713-3cd2-458c-89e1-68995c2c1047",  
      "tag": "BNGRmgfEnDg=",  
      "name": "High Cost Reflections",  
      "maxMemoryPerNodeBytes": 8589934592,  
      "maxQueryMemoryPerNodeBytes": 8589934592,  
      "cpuTier": "BACKGROUND",  
      "maxAllowedRunningJobs": 10,  
      "maxStartTimeoutMs": 300000,  
      "maxRunTimeoutMs": 300000,  
      "engineId": "DATA"  
    },  
    {  
      "id": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",  
      "tag": "HM2D9XElG3U=",  
      "name": "Low Cost Reflections",  
      "cpuTier": "BACKGROUND",  
      "maxAllowedRunningJobs": 10,  
      "maxStartTimeoutMs": 86400000  
    },  
  ]  
}
```



```

{
  "id": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
  "tag": "p22KaFcaB7g=",
  "name": "COPY & OPTIMIZATION Queue",
  "maxMemoryPerNodeBytes": 4294967296,
  "maxQueryMemoryPerNodeBytes": 4294967296,
  "cpuTier": "MEDIUM",
  "maxAllowedRunningJobs": 2,
  "maxStartTimeoutMs": 300000,
  "engineId": "YARN"
},
{
  "id": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "tag": "//gNL3Ta2bY=",
  "name": "Low Cost User Queries",
  "cpuTier": "MEDIUM",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000
},
{
  "id": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "tag": "wa+vYmA73gU=",
  "name": "High Cost User Queries",
  "cpuTier": "MEDIUM",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000
}
]
}

```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a Queue by ID

Retrieve a specific WLM queue by the queue's ID.

Method and URL

```
GET /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue you want to retrieve, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Retrieve a Queue by Name

Retrieve a specific WLM queue by the queue's name.

Method and URL

```
GET /api/v3/wlm/queue/by-name/{name}
```

Parameters

name Path String

Name for the queue you want to retrieve. If the queue name includes special characters for a URL, such as spaces, use URL encoding to replace the special characters with their UTF-8-equivalent characters. For example, "Dremio University" should be `Dremio%20University` in the URL path.

Example: High%20Cost%20Reflections

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/queue/by-name/High%20Cost%20Reflections' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "BACKGROUND",
  "maxAllowedRunningJobs": 10,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

401 Unauthorized

403 Forbidden

404 Not Found

500 Internal Server Error

Update a Queue

Update the specified WLM queue.

Method and URL

Parameters

id Path String (UUID)

Unique identifier of the queue you want to update, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

tag Body String

Unique identifier of the version of the queue to update. Dremio uses the tag to ensure that you are updating the most recent version of the queue.

Example: BNGRmgfEnDg=

name Body String

User-provided name for the queue.

Example: High Cost Reflections

maxMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that all queries running in parallel in a given queue can use per executor node. If you omit the maxMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

maxQueryMemoryPerNodeBytes Body Integer Optional

Total memory (in bytes) that each query in a given queue can use per executor node. If you omit the maxQueryMemoryPerNodeBytes parameter in a PUT request, Dremio removes the existing maxQueryMemoryPerNodeBytes value from the queue. To keep the existing value while making other updates, include the existing maxQueryMemoryPerNodeBytes parameter and value in the PUT request.

Example: 8589934592

cpuTier Body String Optional

Amount of CPU time that threads should get compared to other threads. Default is `MEDIUM`. If you omit the cpuTier parameter in a PUT request, Dremio replaces it with the default value. To keep the existing setting while making other updates, include the existing cpuTier parameter and setting in the PUT request.

Enum: BACKGROUND, LOW, MEDIUM, HIGH, CRITICAL

Example: LOW

maxAllowedRunningJobs Body Integer Optional

Number of queries that are allowed to run in parallel. If you omit the `maxAllowedRunningJobs` parameter in a PUT request, Dremio removes the existing `maxAllowedRunningJobs` value from the queue. To keep the existing value while making other updates, include the existing `maxAllowedRunningJobs` parameter and value in the PUT request.

Example: 100

`maxStartTimeoutMs` Body Integer Optional

Maximum length of time that a query can wait in the queue before it is cancelled, in milliseconds. If you omit the `maxStartTimeoutMs` parameter in a PUT request, Dremio removes the existing `maxStartTimeoutMs` setting from the queue. To keep the existing setting while making other updates, include the existing `maxStartTimeoutMs` parameter and setting in the PUT request.

Example: 300000

`maxRunTimeoutMs` Body Integer Optional

Maximum length of time that a query is allowed to run before it is cancelled, in milliseconds. If you omit the `maxRunTimeoutMs` parameter in a PUT request, Dremio removes the existing `maxRunTimeoutMs` value from the queue. To keep the existing value while making other updates, include the existing `maxRunTimeoutMs` parameter and value in the PUT request.

Example: 300000

`engineId` Body String

Name of the execution engine to which the queue's queries should be routed. If you do not specify an `engineId`, the queue's queries run on any engine that is available at the time of execution.

Example: DATA

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "BNGRmgfEnDg=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "LOW",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}'
```

Response

```
{
  "id": "1990e713-3cd2-458c-89e1-68995c2c1047",
  "tag": "xQh6KNyEjus=",
  "name": "High Cost Reflections",
  "maxMemoryPerNodeBytes": 8589934592,
  "maxQueryMemoryPerNodeBytes": 8589934592,
  "cpuTier": "LOW",
  "maxAllowedRunningJobs": 100,
  "maxStartTimeoutMs": 300000,
  "maxRunTimeoutMs": 300000,
  "engineId": "DATA"
}
```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

500 Internal Server Error

Delete a Queue

Delete the specified WLM queue.

Method and URL

```
DELETE /api/v3/wlm/queue/{id}
```

Parameters

id Path String (UUID)

Unique identifier of the queue that you want to delete, in UTC format.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

Example

Request

```
curl -X DELETE 'https://{hostname}/api/v3/wlm/queue/1990e713-3cd2-458c-89e1-68995c2c1047' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json'
```

Response

No response

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

405 Method Not Allowed

Was this page helpful?

[Previous

Workload Management](/current/reference/api/wlm/)[Next

Rule](/current/reference/api/wlm/rule)

Queue Attributes

Create a Queue

Parameters

Example

Response Status Codes

Retrieve All Queues

Example

Response Status Codes

Retrieve a Queue by ID

Parameters

Example

Response Status Codes

-

Retrieve a Queue by Name

Parameters

Example

Response Status Codes

Update a Queue

Parameters

Example

Response Status Codes

Delete a Queue

Parameters

Example

Response Status Codes

Source:
<https://docs.dremio.com/current/reference/api/wlm/rule/>

Version: current [26.x]

On this page

Rule Enterprise

Use the Workload Management (WLM) API to create, retrieve, update, and delete WLM rules.

The rule object includes a rules array (also called the ruleset). Each object in the rules array represents an individual rule. Dremio processes rules in the order they are listed within the rules array: the highest-priority rule is listed first, and the lowest-priority rule is listed last.

Rule Object

```
{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
```



```

    "name": "Low Cost Reflections",
    "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
    "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
    "acceptName": "Low Cost Reflections",
    "action": "PLACE",
    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "COPY & OPTIMIZATION Rule",
    "conditions": "query_label() in ('COPY','OPTIMIZATION')",
    "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
    "acceptName": "COPY & OPTIMIZATION Queue",
    "action": "PLACE",
    "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Rule Attributes

tag String

Unique identifier of the version of the rule. Dremio changes the tag whenever a rule changes and uses the tag to ensure that PUT requests apply to the most recent version of the rules.

Example: VmqwaZ90VY4=

rules Array of Object

List of rule objects in the Dremio instance.

defaultRule Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule.

Example: `{"name": "All Other Queries", "action": "REJECT", "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}`

Attributes of Objects in the `rules` Array

name String

User-provided name for the rule.

Example: High Cost Reflections

conditions String

Conditions that queries must match to be placed in the queue.

Example: `query_type() = 'Reflections' AND query_cost() >= 30000000`

acceptId String (UUID)

Unique identifier of the queue in which the rule places queries.

Example: 1990e713-3cd2-458c-89e1-68995c2c1047

acceptName Integer

User-provided name for the queue in which the rule places queries.

Example: High Cost Reflections

rejectMessage Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Integer

Action the rule takes for queries that match the rule conditions.

Enum: PLACE, REJECT

Example: PLACE

id String (UUID)

Unique identifier of the rule, in UTC format.

Example: fa1ec87d-923b-414c-9064-e079f39f5c49

Attributes of the `defaultRule` Object

name String

User-provided name for the default rule.

Example: All Other Queries

acceptId String (UUID)

For default rules whose action is ``PLACE``, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Integer

For default rules whose action is ``PLACE``, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Integer

For default rules whose action is ``REJECT``, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action String

Action the default rule takes for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

Create or Update a Rule

Create or update a WLM rule.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To add or update an individual rule, you must include the entire rules array in the request body. It is not necessary to specify the ID for the rule you want to delete in the request URL.

Method and URL

```
PUT /api/v3/wlm/rule
```

Parameters

tag Body String

Unique identifier of the rules instance. Dremio uses the tag to ensure that PUT requests apply to the most recent version of the rules. Omit if you are creating rules for the organization for the first time.

Example: VmqwaZ90VY4=

rules Body Array of Object

List of rule objects in the Dremio instance. To add or update an individual rule, you must include the entire rules array in the request body.

Example:

defaultRule Body Object

Information about the default rule for queries. Dremio applies the default rule to queries that do not meet the conditions for any other rule. To add or update an individual rule, you must include the defaultRule object in the request body.

Example: {"name": "All Other Queries","action": "REJECT","id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"}

Parameters of Objects in the `rules` Array

name Body String

User-provided name for the rule.

Example: DevOps and Engineering

conditions Body String

Conditions that queries must match to be placed in the queue.

Example: is_member('DevOps') OR is_member('Engineering')

acceptId Body String (UUID)

For rules whose action is `PLACE`, the unique identifier of the queue in which the rule should place queries.

Example: b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1

acceptName Body Integer Optional

For rules whose action is `PLACE`, the user-provided name for the queue in which the rule should place queries.

Example: High Cost Reflections

rejectMessage Body Integer

For rules whose action is `REJECT`, a user-provided message for queries that do not match the rule conditions.

Example: Rejected because query does not meet the rule conditions

action Body Integer Optional

Action the rule should take for queries that match the rule conditions. Default is `PLACE`.

Enum: PLACE, REJECT

Example: PLACE

Parameters of the `defaultRule` Object

name Body String

User-provided name for the default rule.

Example: All Other Queries

acceptId Body String (UUID)

For default rules whose action is `PLACE`, the unique identifier of the queue in which the default rule places queries.

Example: a254d63e-9b0e-41be-af4a-1acc5bfe2332

acceptName Body Integer

For default rules whose action is `PLACE`, the user-provided name for the queue in which the rule places queries.

Example: Low Cost User Queries

rejectMessage Body Integer

For default rules whose action is `REJECT`, a user-provided message for queries that do not match any rule conditions.

Example: Rejected because query does not meet any rule conditions

action Body String

Action the default rule should take for queries that do not match the conditions for any other rule.

Enum: PLACE, REJECT

Example: REJECT

id Body String (UUID)

Unique identifier of the default rule.

Example: 8df37560-68c5-45a6-8e1f-4ee2e8438f81

This example request demonstrates how to add a `DevOps and Engineering` rule to the ruleset:

Example Request to Add a Rule

```

curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ]
},

```

```
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'
```

Example Response

```
{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "DevOps and Engineering",
      "conditions": "is_member('DevOps') OR is_member('Engineering')",
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",
      "acceptName": "DevOps and Eng Testing",
      "action": "PLACE",
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    }
  ],
}
```

```

{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

To change the order in which Dremio processes rules, send a PUT request that lists the rules in the desired order within the rules array, with the highest-priority rule listed first, and the lowest-priority rule listed last.

This example request reorders the rules so that the `COPY & OPTIMIZATION Rule` will be the highest-priority rule:

Example Request to Reorder Rules

```

curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "VmqwaZ90VY4=",
  "rules": [
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    }
  ]
}

```



```

},
{
  "name": "High Cost User Queries",
  "conditions": "query_cost() >= 300000000",
  "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
  "acceptName": "High Cost User Queries",
  "action": "PLACE",
  "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
},
{
  "name": "Low Cost User Queries",
  "conditions": "query_cost() < 300000000",
  "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
  "acceptName": "Low Cost User Queries",
  "action": "PLACE",
  "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
}
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}'

```

Example Response

```

{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
    }
  ]
}

```

```

    "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
  },
  {
    "name": "High Cost User Queries",
    "conditions": "query_cost() >= 300000000",
    "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
    "acceptName": "High Cost User Queries",
    "action": "PLACE",
    "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
  },
  {
    "name": "Low Cost User Queries",
    "conditions": "query_cost() < 300000000",
    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

400 Bad Request

401 Unauthorized

403 Forbidden

404 Not Found

409 Conflict

Retrieve All Rules

Retrieve all WLM rules.

Method and URL

```
GET /api/v3/wlm/rule
```

Example

Request

```
curl -X GET 'https://{hostname}/api/v3/wlm/rule' \  
--header 'Authorization: Bearer <dremioAccessToken>' \  
--header 'Content-Type: application/json'
```

Response

```
{  
  "tag": "3uzixTFD134=",  
  "rules": [  
    {  
      "name": "DevOps and Engineering",  
      "conditions": "is_member('DevOps') OR is_member('Engineering')",  
      "acceptId": "b9g7r35c-bda9-e4fb-bagf-9ceaceb9f7c1",  
      "acceptName": "DevOps and Eng Testing",  
      "action": "PLACE",  
      "id": "e4983ad5-cd4b-4b4a-9410-b5c37021ce34"  
    },  
    {  
      "name": "High Cost Reflections",  
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 300000000",  
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",  
      "acceptName": "High Cost Reflections",  
      "action": "PLACE",  
      "id": "falec87d-923b-414c-9064-e079f39f5c49"  
    },  
    {  
      "name": "Low Cost Reflections",  
      "conditions": "query_type() = 'Reflections' AND query_cost() < 300000000",  
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",  
      "acceptName": "Low Cost Reflections",  
      "action": "PLACE",  
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"  
    },  
    {  
      "name": "COPY & OPTIMIZATION Rule",  
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",  
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",  
      "acceptName": "COPY & OPTIMIZATION Queue",  
      "action": "PLACE",  
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"  
    },  
    {  
      "name": "High Cost User Queries",  
      "conditions": "query_cost() >= 300000000",  
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",  
      "acceptName": "High Cost User Queries",  
      "action": "PLACE",  
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"  
    },  
    {  
      "name": "Low Cost User Queries",  
      "conditions": "query_cost() < 300000000",
```

```

    "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
    "acceptName": "Low Cost User Queries",
    "action": "PLACE",
    "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
  }
],
"defaultRule": {
  "name": "All Other Queries",
  "action": "REJECT",
  "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
}
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Delete a Rule

Delete a WLM rule from the ruleset.

In the WLM API, you interact with the ruleset in the rules array rather than individual rules themselves. To delete a rule, send a PUT request that omits the rule from the rules array. It is not necessary to specify the ID for the rule you want to delete in the request URL.

note

The default rule can be updated but cannot be deleted.

Method and URL

```
PUT /api/v3/wlm/rule/
```

This example request demonstrates how to remove the `DevOps and Engineering` rule added in the Creating or Updating a Rule example:

Example

Request

```
curl -X PUT 'https://{hostname}/api/v3/wlm/rule' \
--header 'Authorization: Bearer <dremioAccessToken>' \
```

```
--header 'Content-Type: application/json' \
--data-raw '{
  "tag": "3uzixTFD134=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "fa1ec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY','OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ],
  "defaultRule": {
    "name": "All Other Queries",
    "action": "REJECT",
    "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
  }
}'
```

Response

```

{
  "tag": "VmqlwaZ90VY4=",
  "rules": [
    {
      "name": "High Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() >= 30000000",
      "acceptId": "1990e713-3cd2-458c-89e1-68995c2c1047",
      "acceptName": "High Cost Reflections",
      "action": "PLACE",
      "id": "falec87d-923b-414c-9064-e079f39f5c49"
    },
    {
      "name": "Low Cost Reflections",
      "conditions": "query_type() = 'Reflections' AND query_cost() < 30000000",
      "acceptId": "0dbc50a0-034d-40f6-92f7-ff11eda0c760",
      "acceptName": "Low Cost Reflections",
      "action": "PLACE",
      "id": "dcf15b80-403c-4eba-b600-41ea9319e103"
    },
    {
      "name": "COPY & OPTIMIZATION Rule",
      "conditions": "query_label() in ('COPY', 'OPTIMIZATION')",
      "acceptId": "450ea2a5-9a64-4679-99cb-7b01bf6bba27",
      "acceptName": "COPY & OPTIMIZATION Queue",
      "action": "PLACE",
      "id": "a7f27aea-1e23-4699-8846-51e731c219e9"
    },
    {
      "name": "High Cost User Queries",
      "conditions": "query_cost() >= 30000000",
      "acceptId": "c2917cce-b566-4c6a-be63-2e28488a6928",
      "acceptName": "High Cost User Queries",
      "action": "PLACE",
      "id": "880d84a2-548d-4040-b6ba-a5371e87aecf"
    },
    {
      "name": "Low Cost User Queries",
      "conditions": "query_cost() < 30000000",
      "acceptId": "a254d63e-9b0e-41be-af4a-1acc5bfe2332",
      "acceptName": "Low Cost User Queries",
      "action": "PLACE",
      "id": "c0fa6e0b-e479-497b-846a-ad543009a309"
    }
  ],
  "defaultRule": {
    "name": "All Other Queries",
    "action": "REJECT",
    "id": "8df37560-68c5-45a6-8e1f-4ee2e8438f81"
  }
}

```

Response Status Codes

200 OK

401 Unauthorized

404 Not Found

409 Conflict

500 Internal Server Error

Was this page helpful?

[Previous

Queue](/current/reference/api/wlm/queue)

Rule Attributes

Create or Update a Rule

Parameters

Example Response

Example Response

Response Status Codes

Retrieve All Rules

Example

Response Status Codes

Delete a Rule

Example

Response Status Codes