CDP Public Cloud / Data Catalog

Profilers in Compute Cluster Enabled Environments

Date published: 2019-11-14
Date modified: 2025-10-17

CLOUDERA

https://docs.cloudera.com/

Legal Notice

Contents

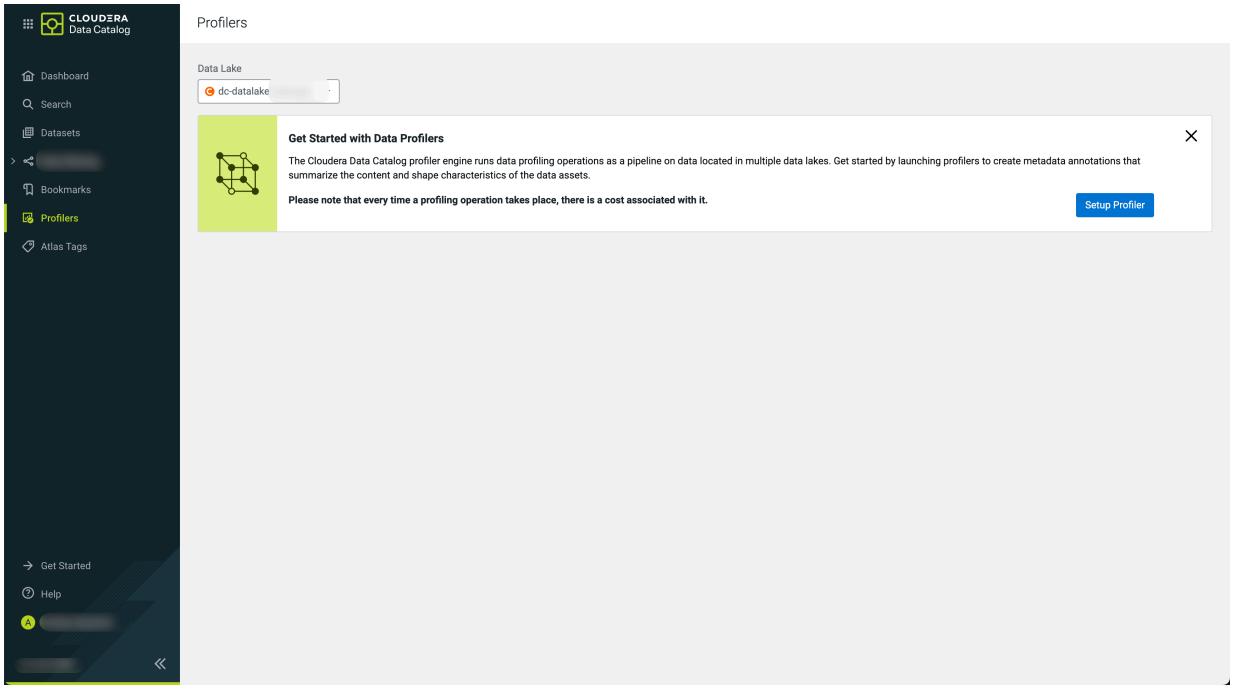
Launching profilers in Compute Cluster enabled environments

In Compute Cluster enabled environments, after you set up the Kubernetes profiler node group, the Profiler Launcher Service (PLS) keeps checking the availability of the node group automatically. Once the node group is ready, the PLS provisions the selected profilers by starting CRON jobs in the Kubernetes node groups.

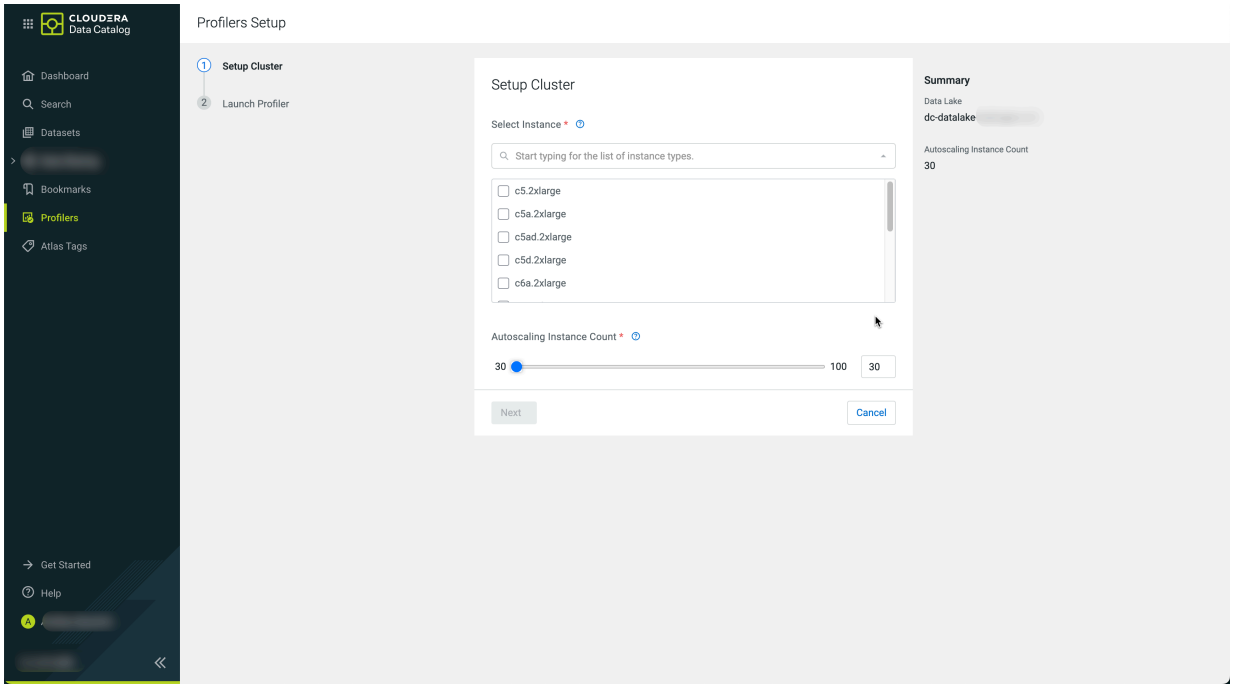
Note: You must be a Power User to launch a profiler cluster.

How to launch the profiler for Compute Cluster enabled environments

1. On the **Profilers** page, select the data lake from which you want to launch the profiler cluster.
2. Click Setup Profiler, to start the profiler cluster setup.



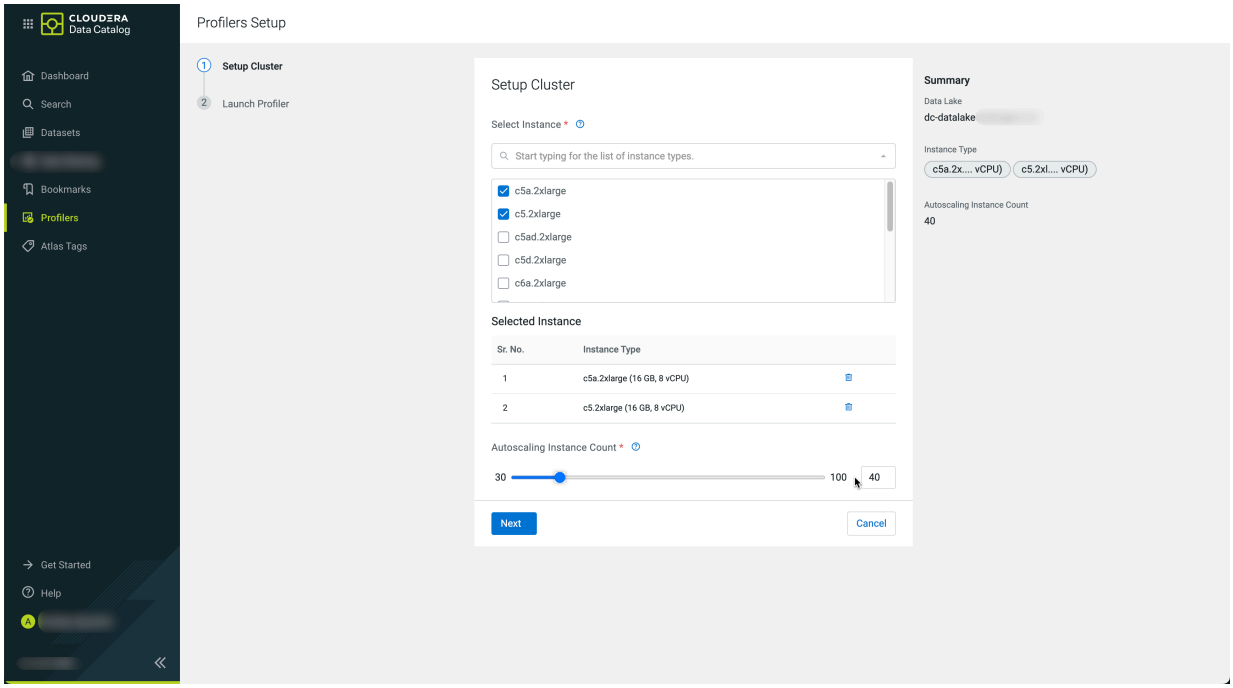
3. In **Setup Cluster**, search for the required instance types:



The available instance types depend on the cloud provider of the underlying environment. Choose from them based on your performance and cost requirements.

Note: For more information, see Amazon EC2 Instance types or Azure Virtual Machine series.

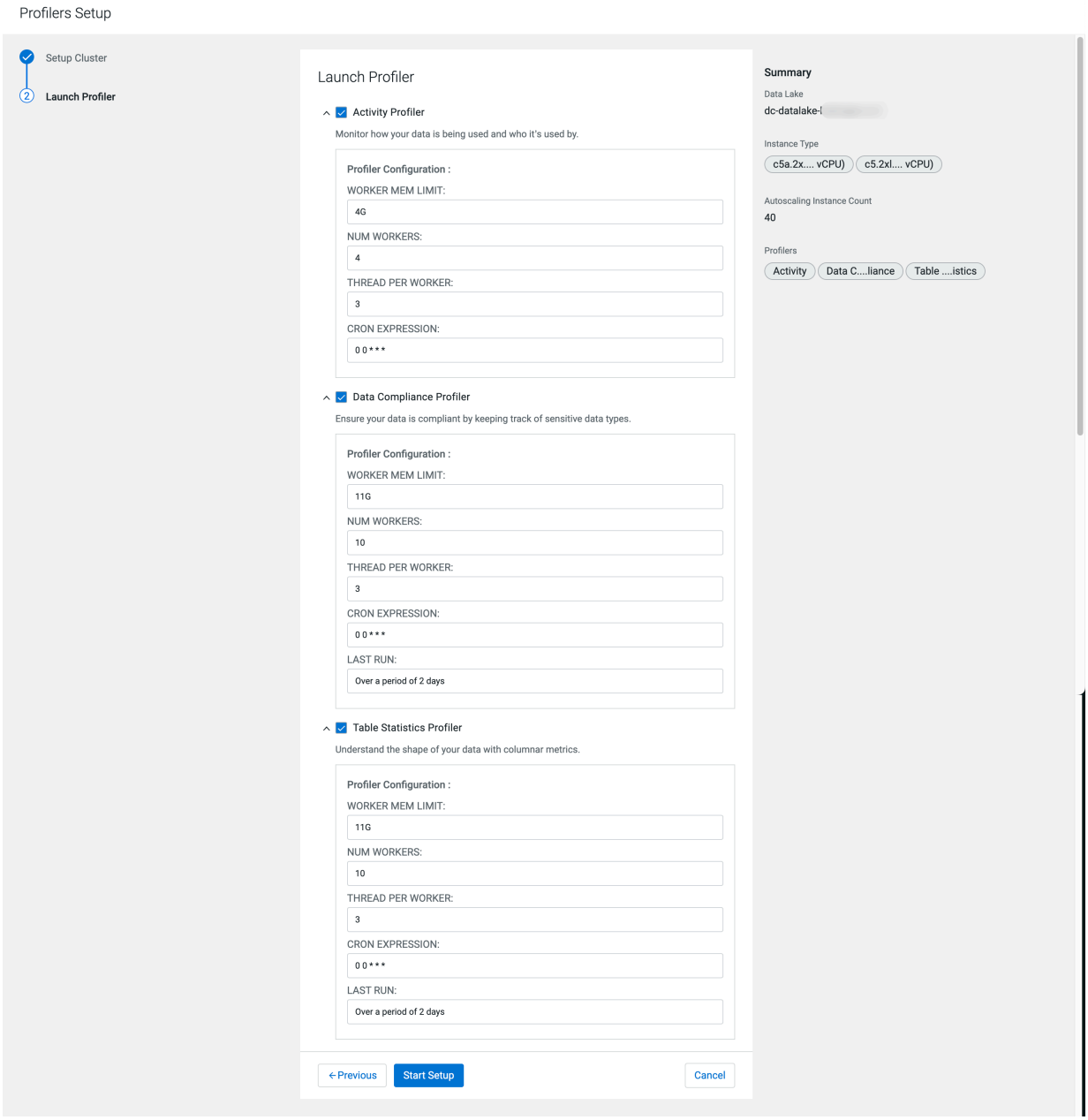
4. Select your required instances and set the Autoscaling instance count to define maximum number of workers. The underlying Apache Spark service will manage the actual number of used instances based on workload.



5. Click Next.

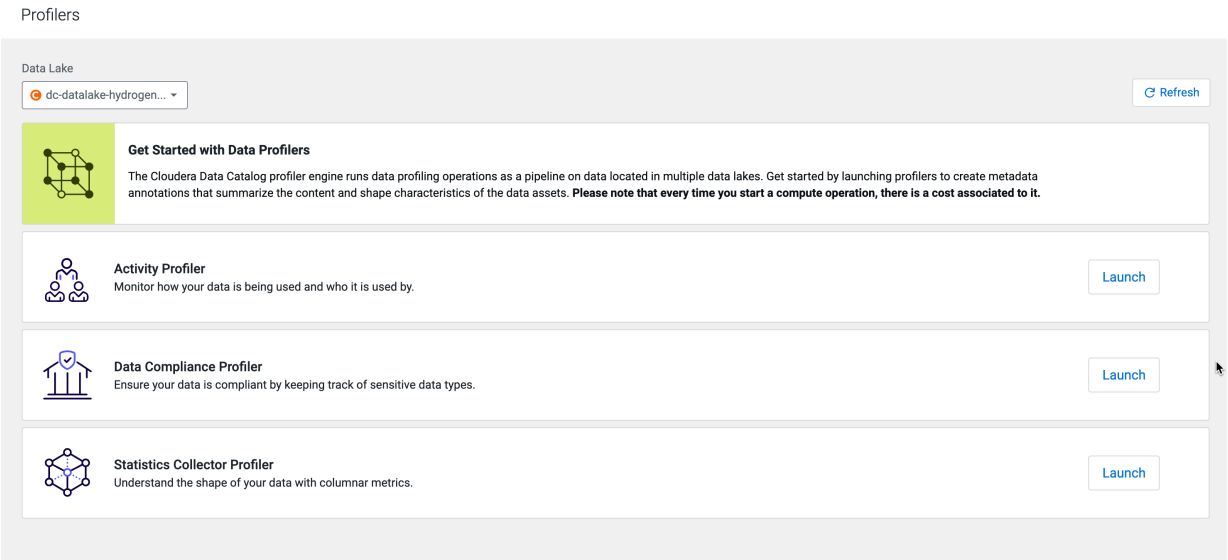
6. Select the necessary profilers to be launched.

Note: Profilers can be launched later as well. Also, their configuration can be changed after launching them.



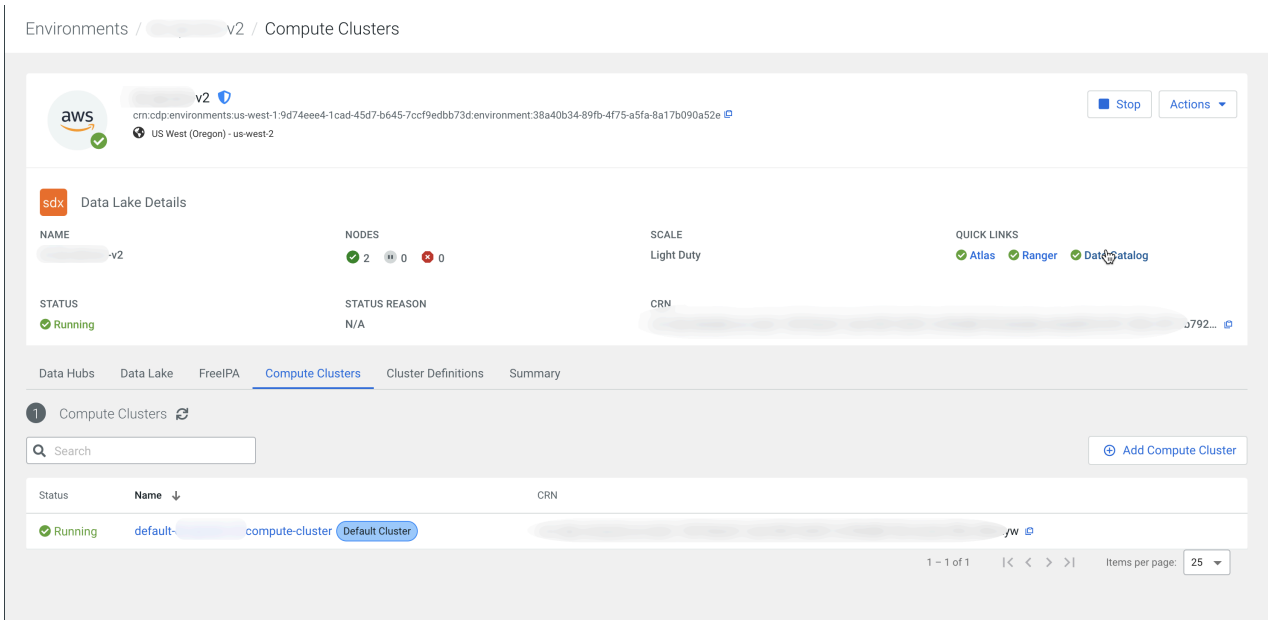
7. Once the cluster is ready to accept Kubernetes profiler jobs, you can start the individual profilers by clicking Launch. If the profiler jobs were scheduled earlier, they will be automatically assigned to the finished Kubernetes node group.

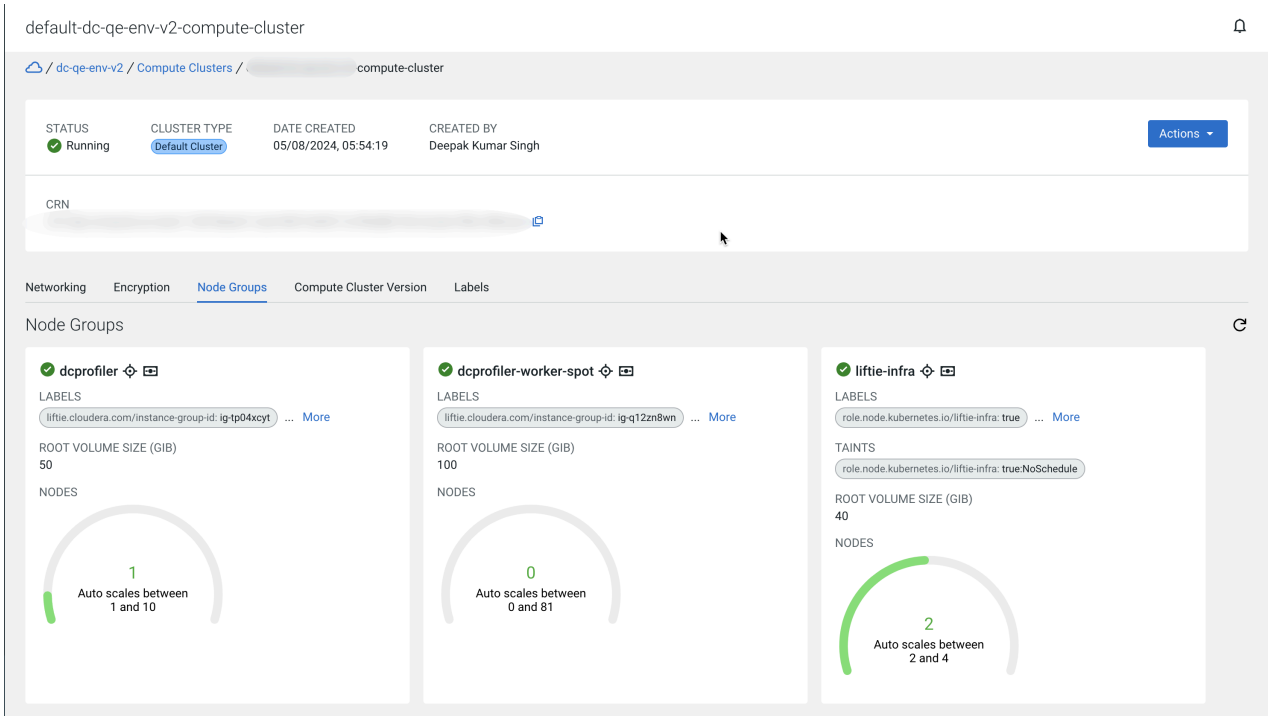
Note: The readiness of the Kubernetes node group can be checked in Cloudera Management Console Environments <***YOUR_ENVIRONMENT***> Compute Clusters . The worker node group is created by the Liftie service. The expected setup time is around 15 to 30 minutes.



Verifying the profiler cluster for Compute Cluster enabled environments

As a final step, you can verify that the node group is ready for the profiler jobs under the Cloudera Management Console Environments Compute Clusters Node Groups pane.





Launching profilers using the command-line

Cloudera Data Catalog supports launching profilers using the Command-Line Interface (CLI) option.

The CLI is one executable and does not have any external dependencies. You can execute some operations in the Cloudera Data Catalog service using the Cloudera CLI commands.

Users must have valid permissions to launch profilers on a data lake.

For more information about the access details, see Prerequisites to access Cloudera Data Catalog.

Prerequisites

You must have the following entitlement granted to use this feature:

DATA_CATALOG_ENABLE_API_SERVICE

For more information about the Cloudera command-line interface and setting up the same, see Cloudera CLI.

The Cloudera Data Catalog CLI

In your Cloudera CLI environment, enter the following command to get started in the CLI mode.

cdp datacatalog --help

This command provides information about the available commands in Cloudera Data Catalog for Cloudera on cloud 7.2.18. and earlier versions.

The output is displayed as:

```
NAME
datacatalog
DESCRIPTION
Cloudera Data Catalog Service is a web service, using this service user can
 execute operations like launching profilers in Data Catalog.
AVAILABLE SUBCOMMANDS
```

```
launch-profilers
```

Parameters for profiler launch command

You get additional information about this command by using:

cdp datacatalog launch-profilers --help

```
NAME
launch-profilers -
DESCRIPTION
Launches DataCatalog profilers in a given datalake.
```

```
NAME
       launch-profilers - Launches DataCatalog profilers in a given datalak
e.

DESCRIPTION
       Launches DataCatalog profilers in a given datalake.

SYNOPSIS

            launch-profilers
          --datalake <value>
          [--enable-ha | --no-enable-ha]
          [--profilers <value>]
          [--instance-types <value>]
          [--max-nodes <value>]
          [--cli-input-json <value>]
          [--generate-cli-skeleton]

OPTIONS
       --datalake (string)
          The CRN of the Datalake.

       --enable-ha | --no-enable-ha (boolean)
          Enables High Availability (HA) for datacatalog profilers (default
          value is false). The High Availability (HA) Profiler cluster
          provides failure resilience and scalability but incurs additional
          cost.
       --profilers (array)
          List of profiler names that need to be launched. (Applicable only
          for compute cluster enabled environments).

       Syntax:

          "string" "string" ...

       --instance-types (array)
          List of instance types to be used for the auto-scaling node group
          setup (Applicable only for compute cluster enabled environments).

       Syntax:

          "string" "string" ...
       --max-nodes (integer)
          Maximum number of nodes that can be spawned inside the auto-scal
ing
          node group, in the range of 30 to 100 (both inclusive). (Applicabl
e
          only for compute cluster enabled environments).
```

```
       --cli-input-json (string)
          Performs service operation based on the JSON string provided. The
          JSON string follows the format provided by --generate-cli-skelet
on.
          If other arguments are provided on the command line, the CLI value
s
          will override the JSON-provided values.
       --generate-cli-skeleton (boolean)
          Prints a sample input JSON to standard output. Note the specified
          operation is not run if this argument is specified. The sample i
nput
          can be used as an argument for --cli-input-json.
OUTPUT
       success -> (boolean)
          Status of the profiler launch operation.


FORM FACTORS
       public
```

Note:

- The following parameters are only applicable to Compute Cluster environments (they are ignored in VM-based environments):
  - --profilers ***VALUE***
  - --instance-types ***VALUE***
  - --max-nodes ***VALUE***

Parameters for profiler delete command

You get additional information about this command by using:

cdp datacatalog delete-profiler --help

```
NAME
       delete-profiler - Deletes DataCatalog profiler in a given datalake.
DESCRIPTION
       Deletes DataCatalog profiler in a given datalake.

SYNOPSIS
            delete-profiler
          --datalake <value>
          [--cli-input-json <value>]
          [--generate-cli-skeleton]

OPTIONS
       --datalake (string)
          The CRN of the Datalake.

       --cli-input-json (string)
          Performs service operation based on the JSON string provided. The
          JSON string follows the format provided by --generate-cli-skeleton
.
          If other arguments are provided on the command line, the CLI val
ues
          will override the JSON-provided values.

       --generate-cli-skeleton (boolean)
          Prints a sample input JSON to standard output. Note the specified
          operation is not run if this argument is specified. The sample inp
ut
          can be used as an argument for --cli-input-json.
```

```
OUTPUT
FORM FACTORS
       public
```

Launching the profiler

You can use the following CLI command to launch the data profiler:

```
cdp datacatalog launch-profilers --datalake [***DATALAKE CRN***]
```

Example:

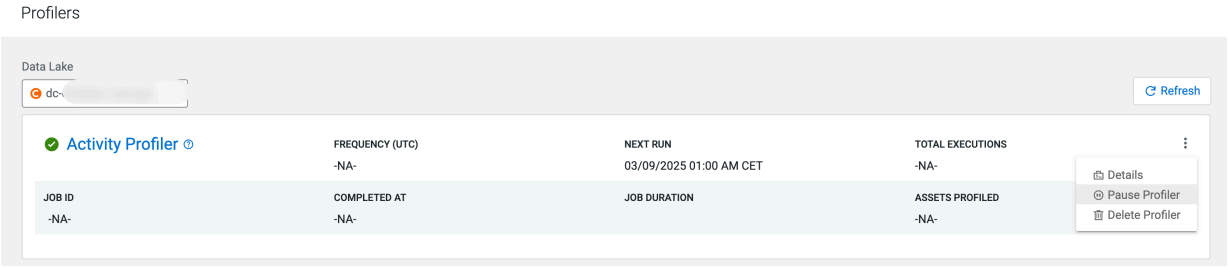
```
cdp datacatalog launch-profilers --datalake crn:cdp:data
lake:DATACENTERNAME:c*****b-ccce-4**d-a**1-8********8:datalake:4*****5e-c**
1-4**2-8**e-1********2
{
    "success": true
}
```

Enabling or disabling profilers in Compute cluster enabled environments

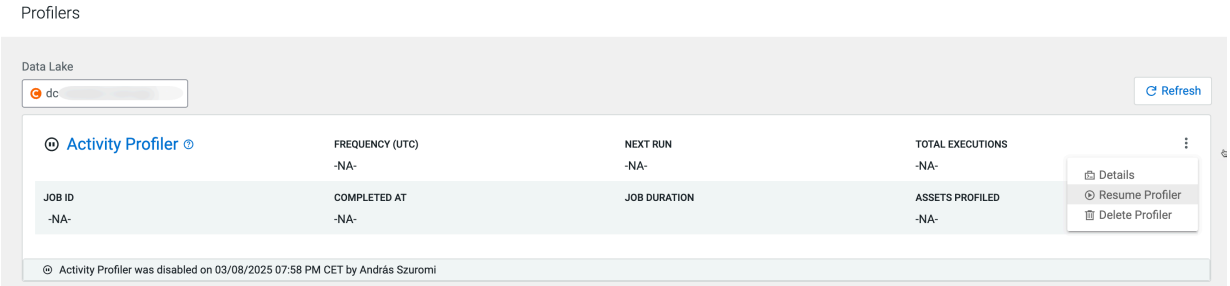
Profilers can be temporarily paused to save resources.

Procedure

1. Go to **Profilers**.
2. Click ⋮ > Pause Profiler.



3. Click Confirm.
4. You can click Resume Profiler to continue using it.

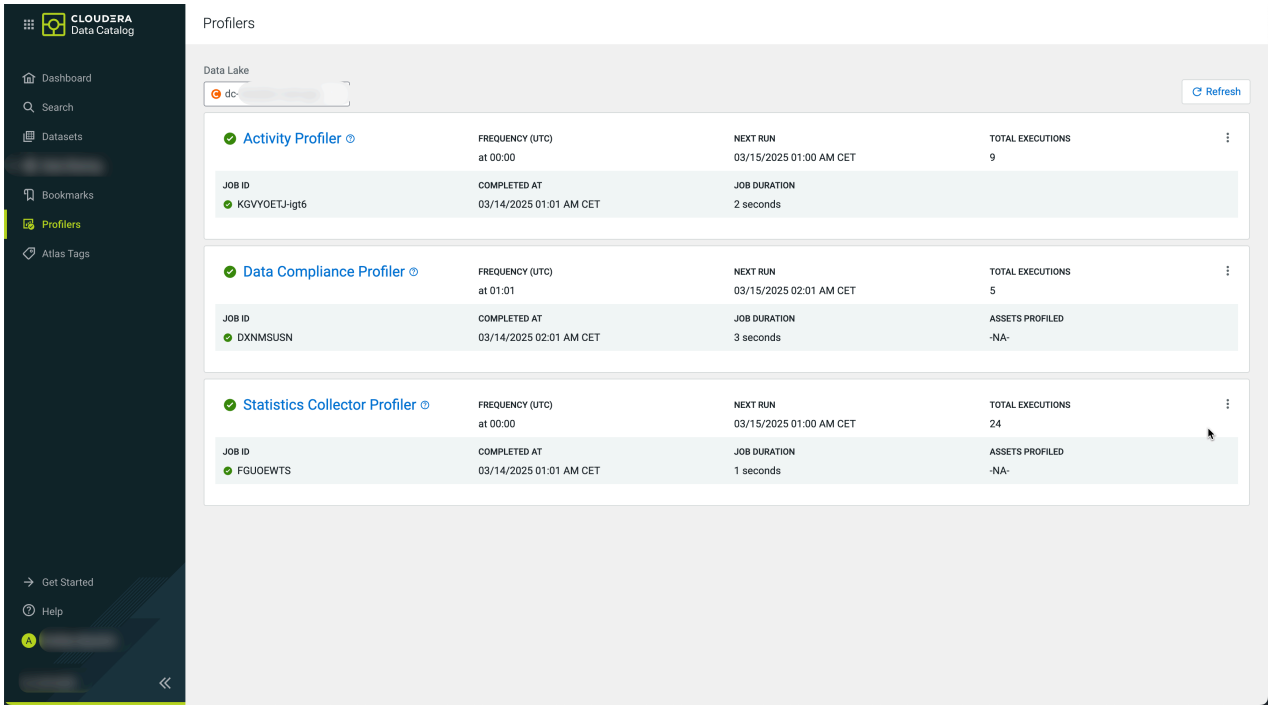


Tracking profiler jobs in Compute cluster enabled environments

In Profilers, you can see the status and statistics of your profilers.

Under Profilers , you can have an overview of your profiler since their launch and some basic information of the last jobs. Use this page to quickly check if your profiler jobs are failing.

Figure 1: Profiling jobs in a Compute Cluster enabled environment



For each profiler, you can view the details about:

- **Profiler** type
- Profiler **Status** for the last job (as an icon , ,)
- **Frequency (UTC)**
- **Next Run** (in your local timezone)
- **Total Executions** (since the launch of the profiler)
- Job status is marked with icons (, ,)
  - Running (Successfully launched)
  - Paused
  - Creation in Progress
- **JOB ID** of the last job
- **COMPLETED AT**
- **JOB DURATION** (of the last job)
- **ASSETS PROFILED** (by the last job)

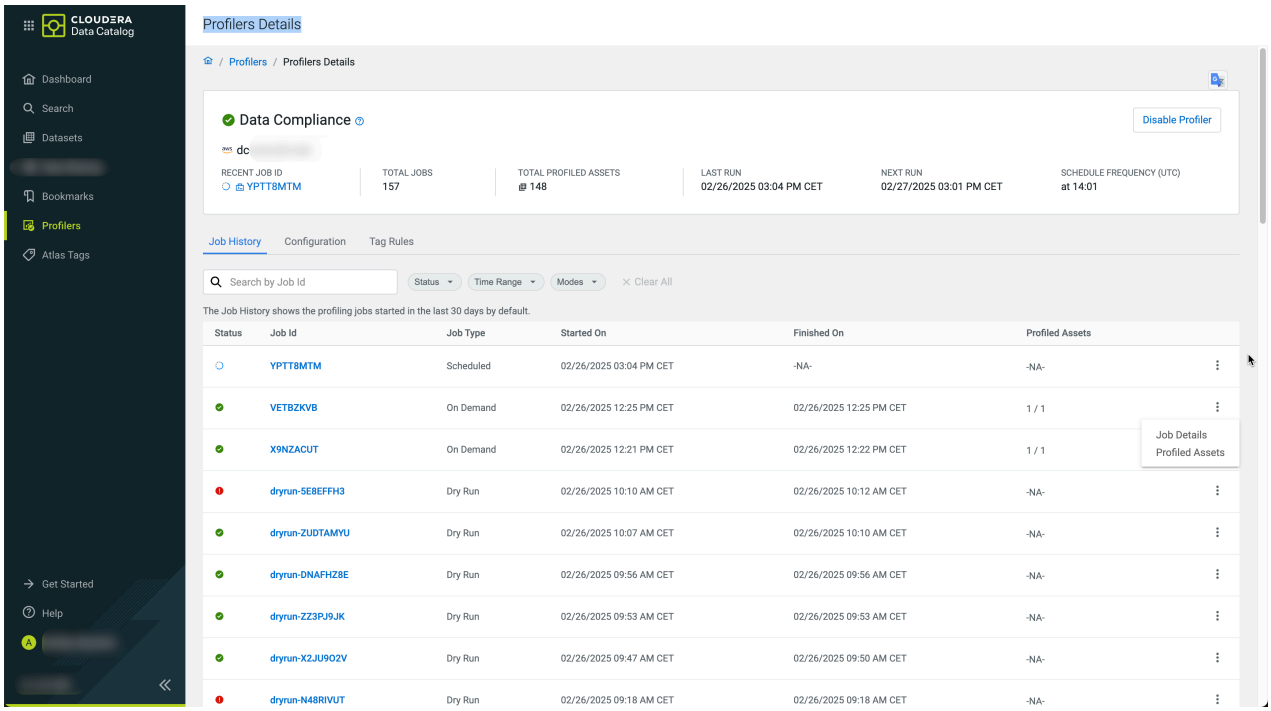
Note: Not available for the Activity Profiler.

Using this data can help you to troubleshoot failed jobs or even understand how the assets were profiled and other pertinent information that can help you to manage your profiled assets.

Click ⋮ >Details to gain more information about your profiler jobs in Profilers Profilers Details .

Use the following filters to screen your profiler jobs:

- Status:
  - Failed
  - Running
  - Finished
- Time Range
- Modes[1]
  - **Dry Run**[2]
  - **On Demand**
  - **Scheduled**

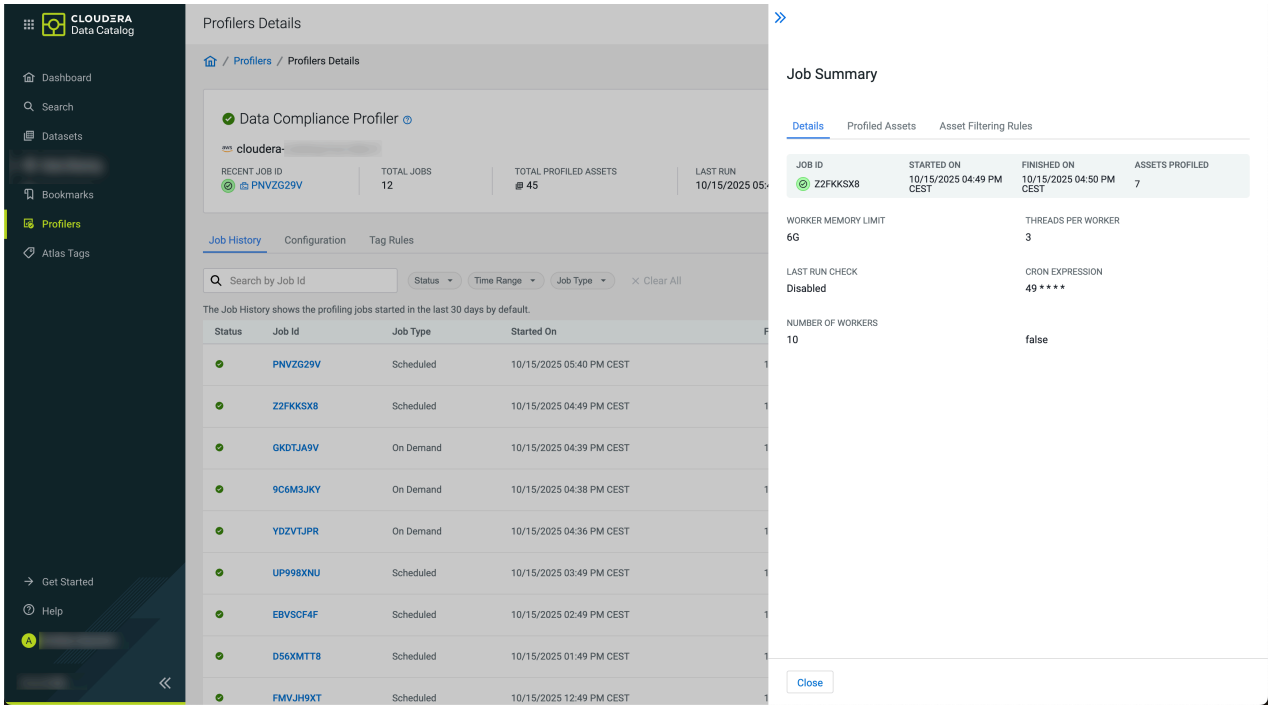


By clicking ⋮ by the individual jobs in **Profilers Details**, you can drill further down to **Job Summary** and **Profiled Assets**.

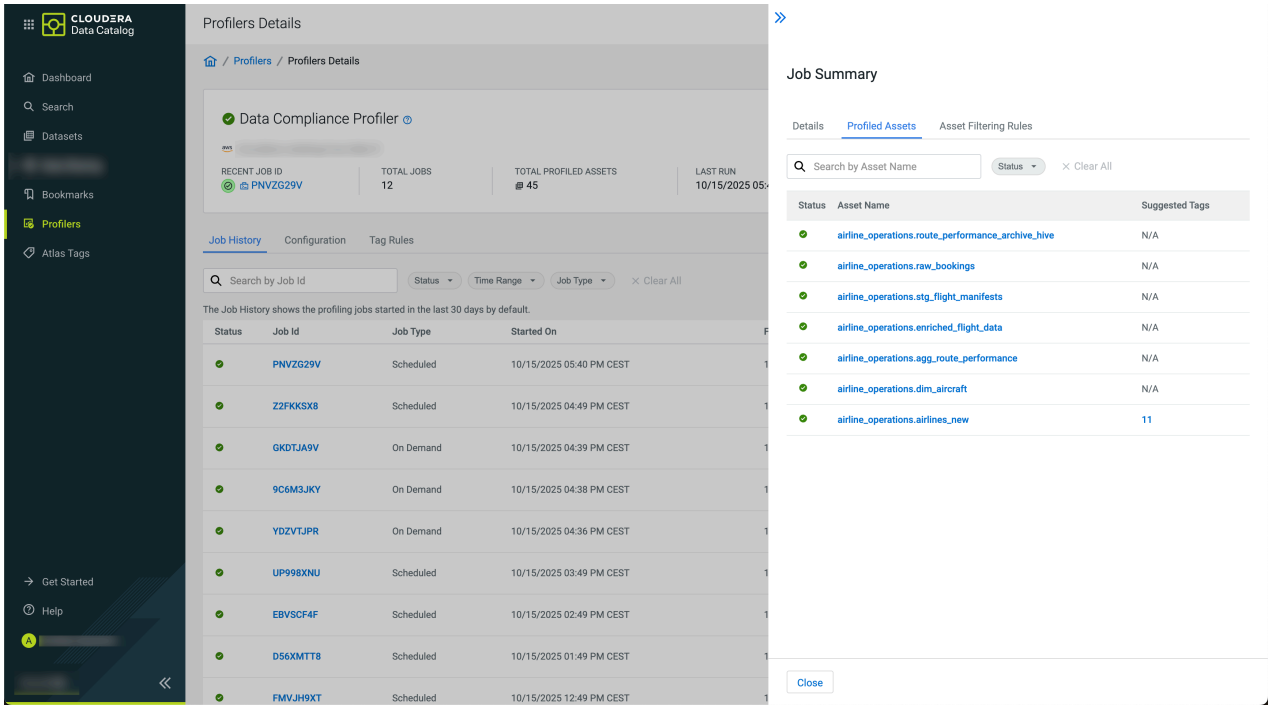
The **Job Summary** shows you the specific configuration applied for that particular job run.

[1] Only available for the Data Compliance and Statistics Collector profilers.
[2] Only available for the Data Compliance profiler.

The Dry Run mode refers to the first on-demand profiler jobs, triggered to validate custom tag rules. These test the tag rule on a subset of entities from the data lake.

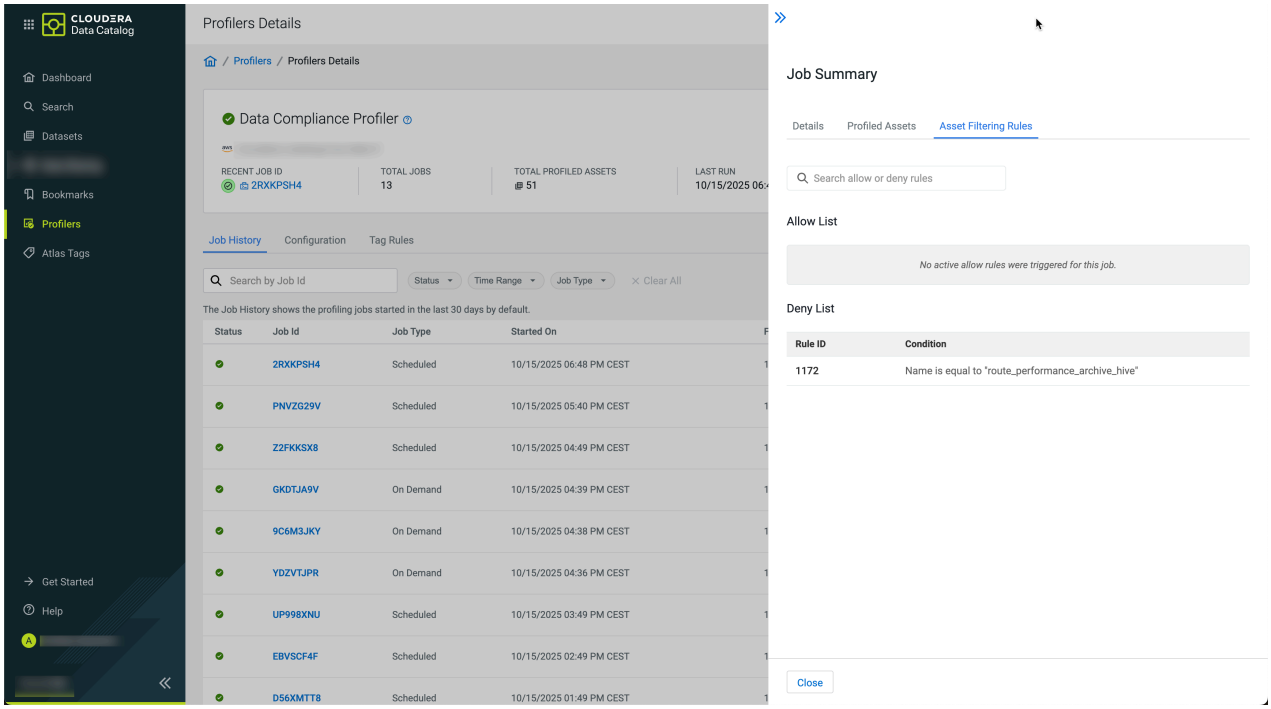


The **Profiled Assets** not only gives you a list of entities that were selected by your profiler to be profiled, but it let's you filter them.



Note: Hovering over the skipped assets shows the reason for not including the particular asset.

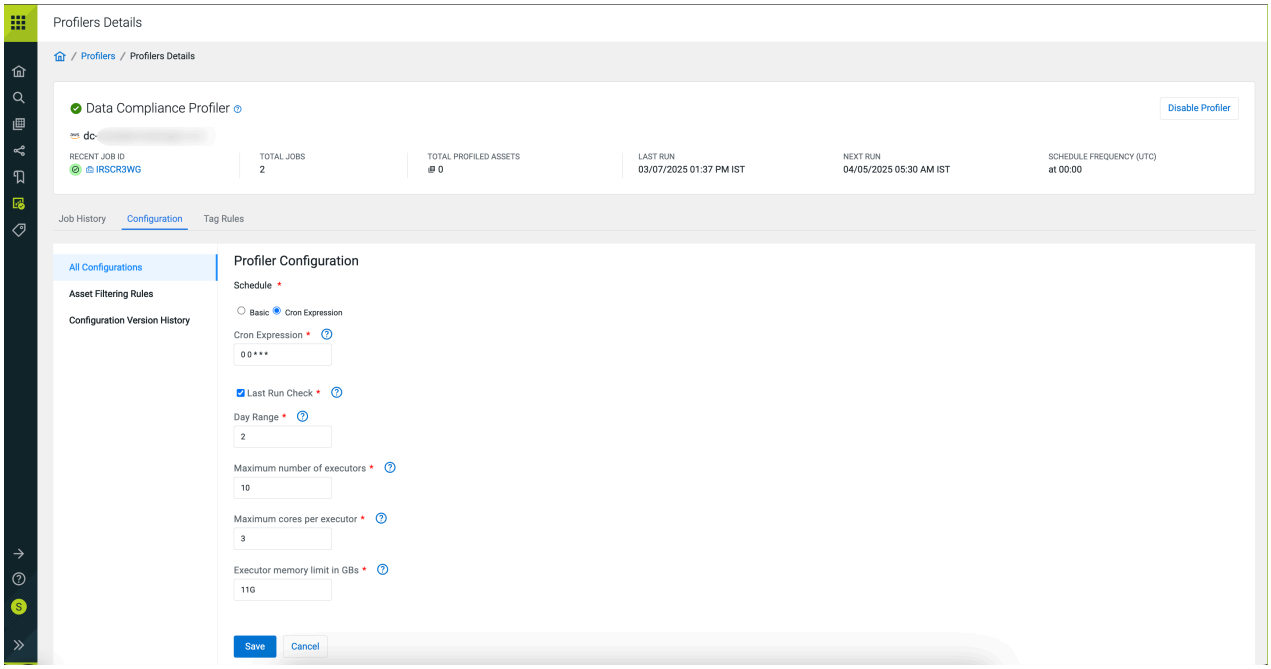
The **Asset Filtering Rules** tab summarizes the active filtering rules for your profiling job.



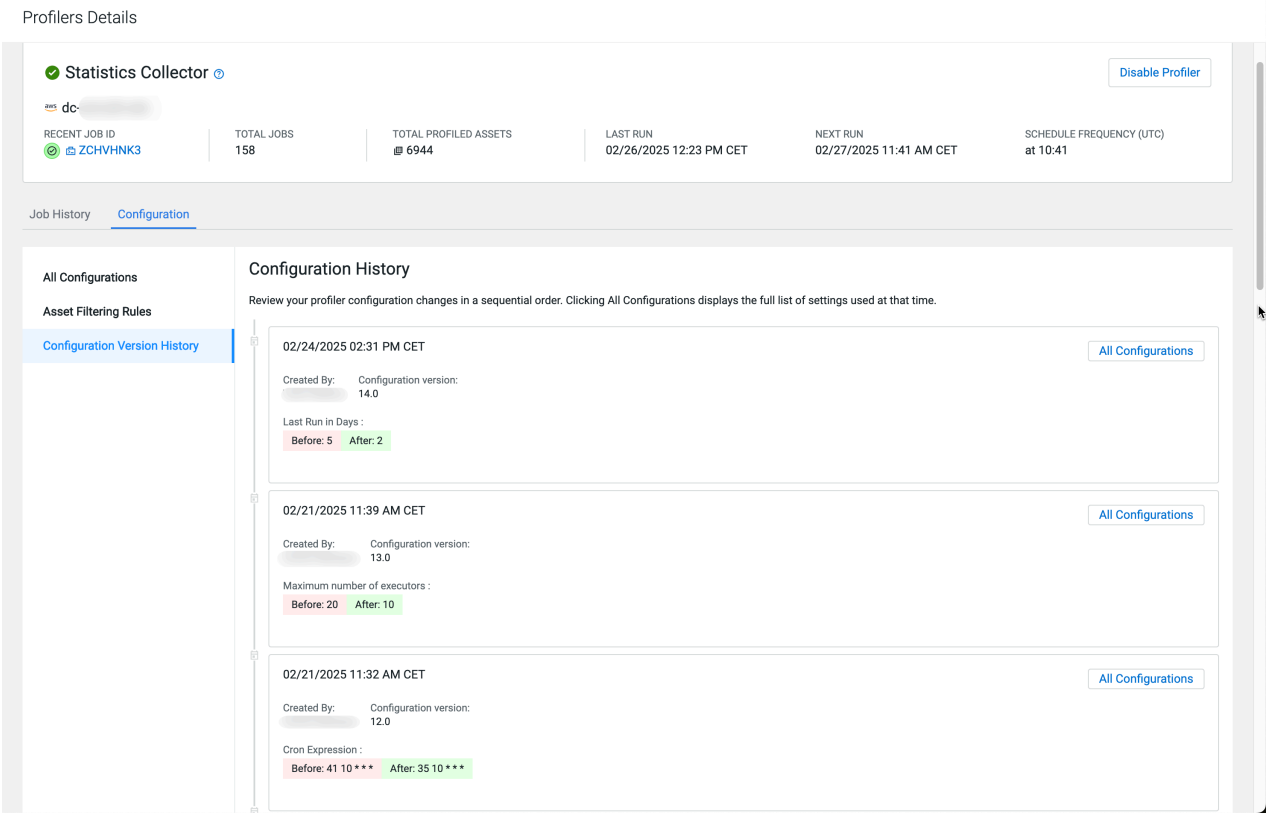
Viewing profiler configurations in Compute cluster enabled environments

You can check the configuration and its changes for your profiles in Profilers > Profiler Details > Configuration.

In Profilers Profiler Details Configuration All Configurations you can set the scheduling and resources of your profilers.



Configuration Version History lets you check your changes to your settings.



Clicking **All Configurations** shows all settings at the time, including the unchanged options.

Configuration version 14.0

Profiler Configuration

Cron Expression : 41 10 * * *	Cron Expression : 41 10 * * *
Last Run in Days : 5	Last Run in Days : 2
Last Run Enabled : true	Last Run Enabled : true

Executor Configurations

Maximum number of executors : 20	Maximum number of executors : 20
Maximum cores per Executor : 3	Maximum cores per Executor : 3
Executor memory limit in GBs : 11G	Executor memory limit in GBs : 11G

Close

Configuring the Activity Profiler

Configure the scheduling and the available resources for your profiler.

Procedure

1. Go to **Profilers** and select your data lake.
2. Go to Profilers Activity Profiler Profiler Details Configuration All Configurations
3. Select a schedule to run profiler using either UNIX Cron Expression or the Basic scheduler.

Note: Both the Basic and Cron Expression scheduler (Unix in Compute Cluster enabled environments cron jobs) use the UTC timezone instead of the local timezone of the user.

Figure 2: Profiler schedule with cron expression

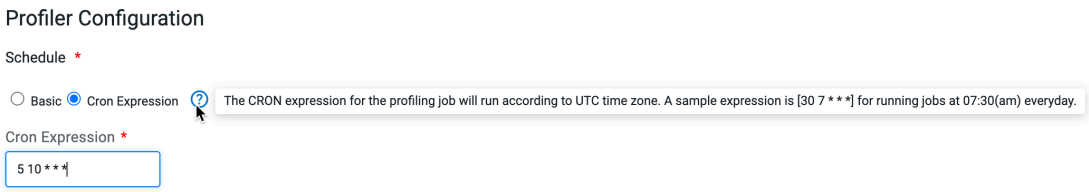


Figure 3: Profiler schedule with natural language

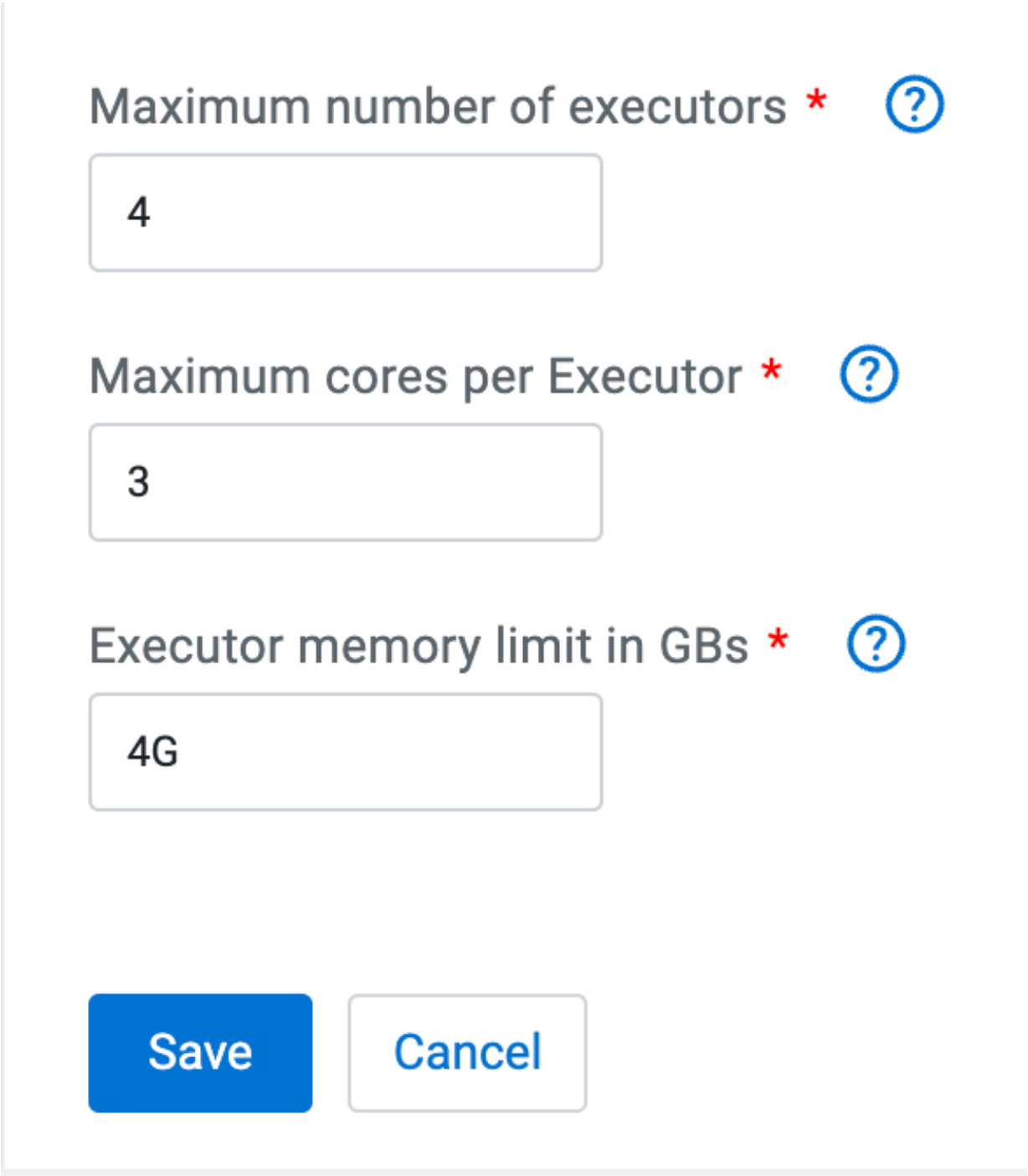
Note:

Compute Cluster based profilers might hang if the underlying AWS cloud provider environment cannot provide the necessary memory for the executor instances. In this case, reconfigure your executors with 4-5 GB memory in Profiler Details Configuration .

4. Continue with resource settings:
   a) Set the Maximum number of executors

      Indicates the number of processes that are used by the distributed computing framework. The recommended value is at least four executors.
   b) Set the Maximum cores per executor

      Indicates the maximum number of cores that can be allocated to an executor.
   c) Set the Executor memory limit in GBs



5. Click Save to apply the configuration changes to the selected profiler.

Configuring the Data Compliance profiler

You can configure the scheduling and the available resources for your profiler.

Procedure

1. Go to **Profilers** and select your data lake.
2. Go to Profilers Data Compliance Profiler Details Configuration All Configurations
3. Select a schedule to run profiler using either UNIX Cron Expression or the Basic scheduler.

   Note: Both the Basic and Cron Expression scheduler (Unix in Compute Cluster enabled environments cron jobs) use the UTC timezone instead of the local timezone of the user.

Figure 4: Profiler schedule with cron expression

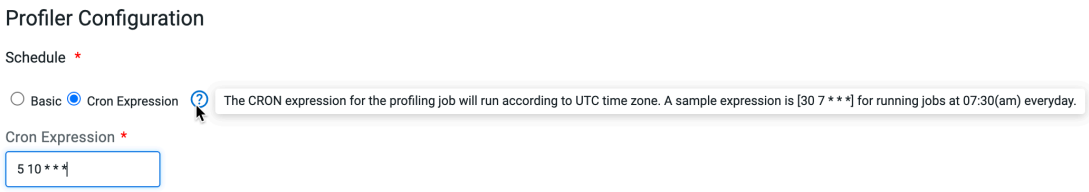
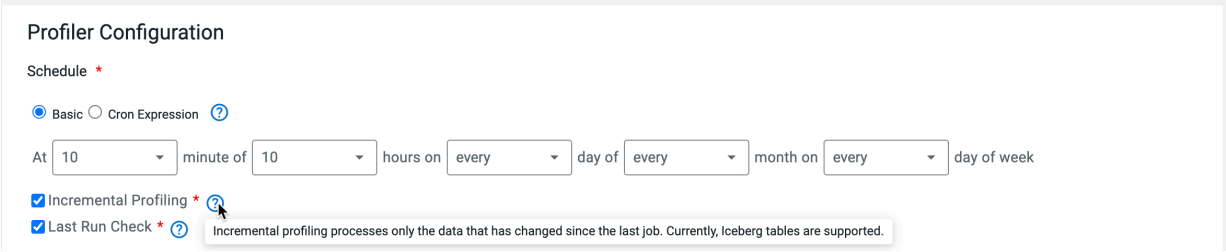


Figure 5: Profiler schedule with natural language



4. Select Incremental Profiling when needed.

   Using Incremental Profiling can decrease the compute resources and the time needed for the profiling job by processing only the information (only Iceberg tables) updated or added since previous job.

   Using Incremental Profiling, you can refine the results from the Last Run Check. Incremental Profiling checks the data (rows) in assets, while Last Run Check filters complete assets.
5. Select Last Run Check and set a period in Day Range if needed.

   Note:

   The Last Run Check enables profilers to avoid profiling the same asset on each scheduled run.

   If you have scheduled a cron job, for example set to start in about an hour, and have enabled the Last Run Check configuration for two days, this setup ensures that the job scheduler filters out any asset which was already profiled in the last two days.

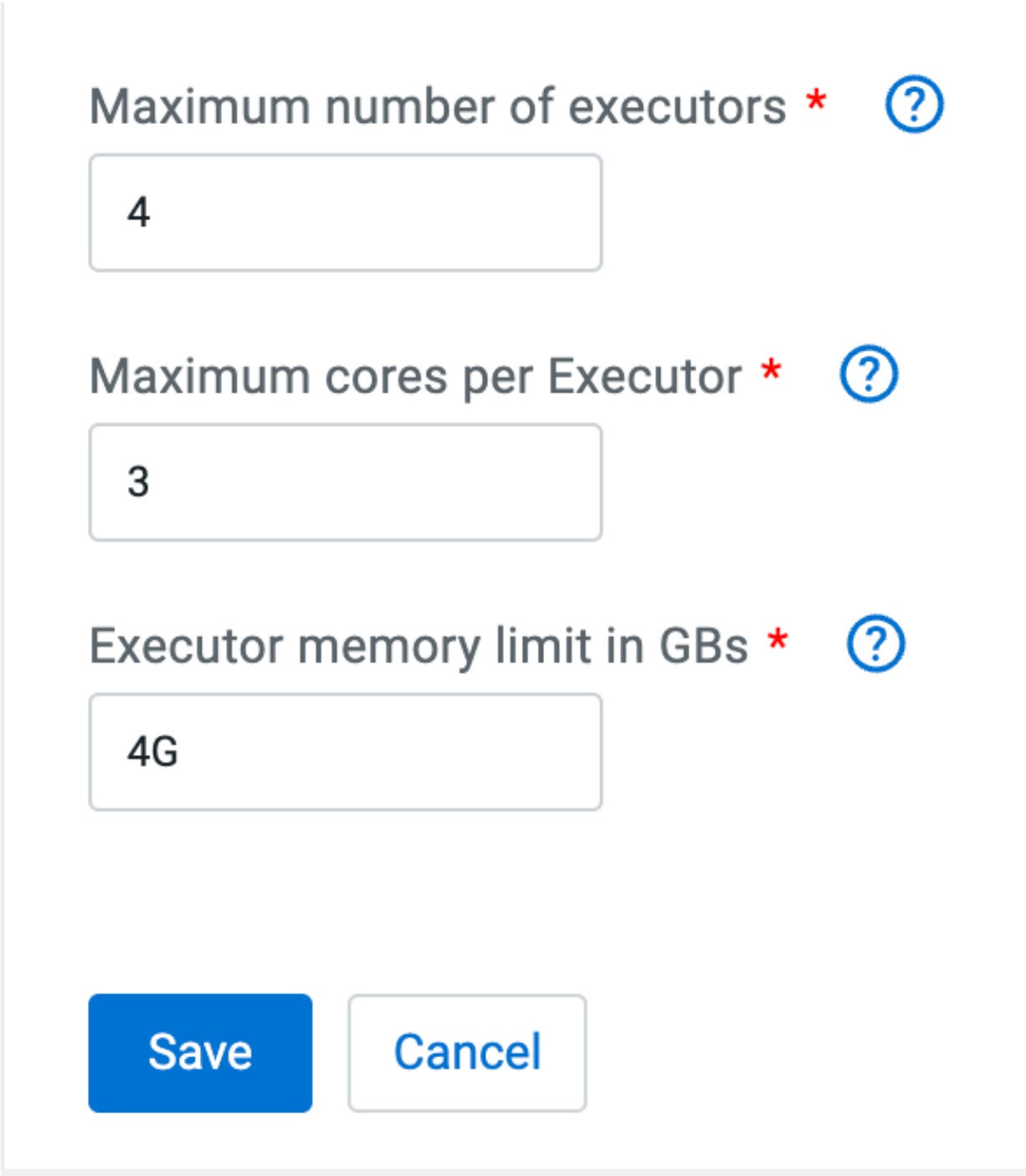
   If the Last Run Check configuration is disabled, assets will be picked up for profiling as per the job cron schedule, honoring the asset filter rules.

   The Last Run Check precedes Incremental Profiling.

6. Continue with resource settings:
   a) Set the Maximum number of executors

      Indicates the number of processes that are used by the distributed computing framework. The recommended value is at least 10 executors.
   b) Set the Maximum cores per executor

      Indicates the maximum number of cores that can be allocated to an executor.
   c) Set the Executor memory limit in GBs



Note:

Compute Cluster based profilers might hang if the underlying AWS cloud provider environment cannot provide the necessary memory for the executor instances. In this case, reconfigure your executors with 4-5 GB memory in Profiler Details Configuration .

7. Click Save to apply the configuration changes to the selected profiler.
8. Add **Asset Filtering Rules** as needed to customize the selection and deselection of assets which the profiler profiles.

Note:

- Profiler configurations apply to both scheduled and on-demand profiler jobs.
- Asset filtering rules apply to assets, such as tables, and not to complete databases.
- Multiple asset filtering rules are evaluated together as if connected by the OR operator.
- In Compute Cluster environments, you cannot enable conflicting Allow and Deny list rules at the same time. Enabling conflicting rules results in an error message.

a) Set your **Deny List** and **Allow-list**.

The profiler will skip profiling assets that meet any criteria in the **Deny List** and will include assets that meet any criteria in the **Allow List**.

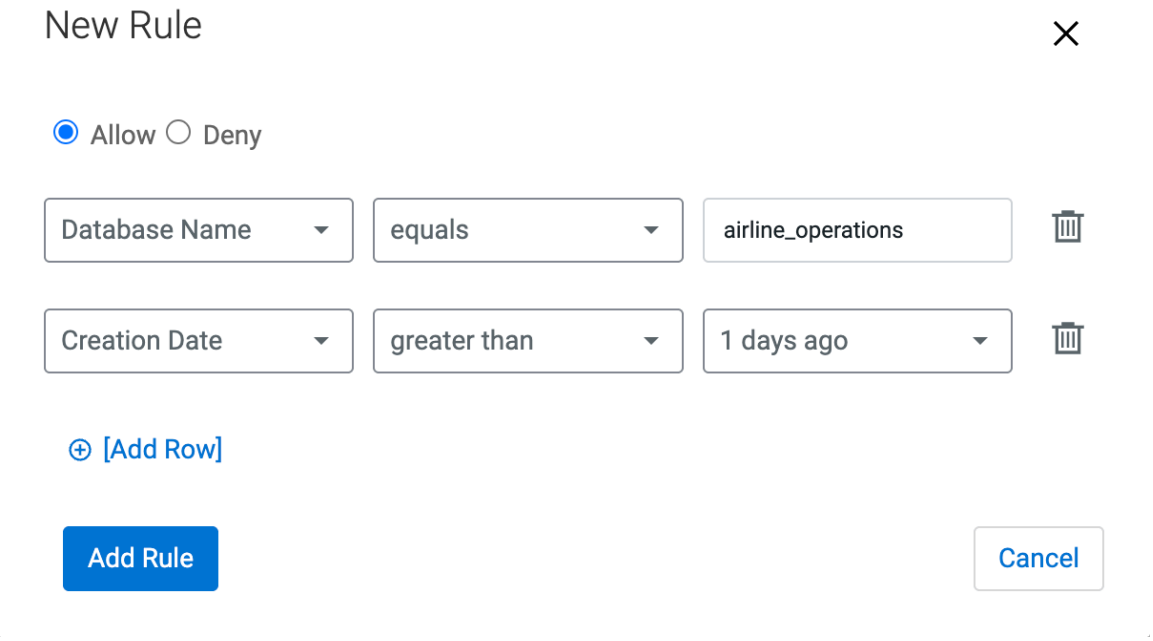
1. Click Add New Rule to define new rules.
2. Use the radio buttons to define your new rule for the Allow or Deny List.
3. Select the key from the drop-down list and the relevant operator. You can select from the following:

Key	Operator
Database name	• equals • starts with • ends with
Name (of asset)	• equals • contains • starts with • ends with
Owner (of asset)	

Key	Operator
Creation date[3]	• greater than • less than

Note: **Name** refers here to the actual name of the asset and not to its **Qualified Name**.

4. Enter the value corresponding to the key. For example, you can enter a string as mentioned in the previous example.
5. Click Add Rule. Once a rule is added (enabled by default), you can toggle the state of the new rule to enable it or disable it as needed.



Note: You can check the list of assets impacted by your rule by clicking ⋮ > Affected Assets.

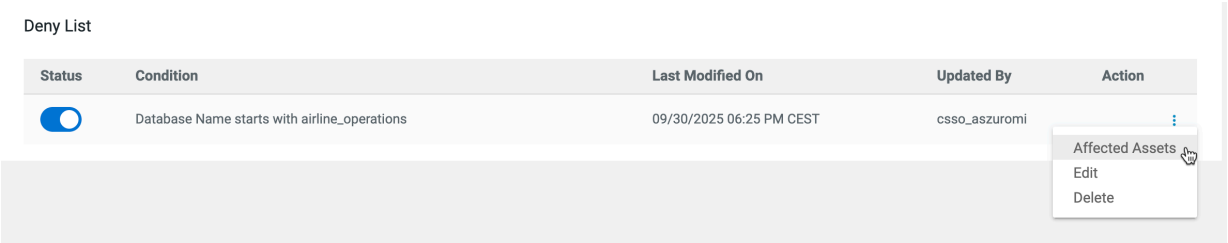
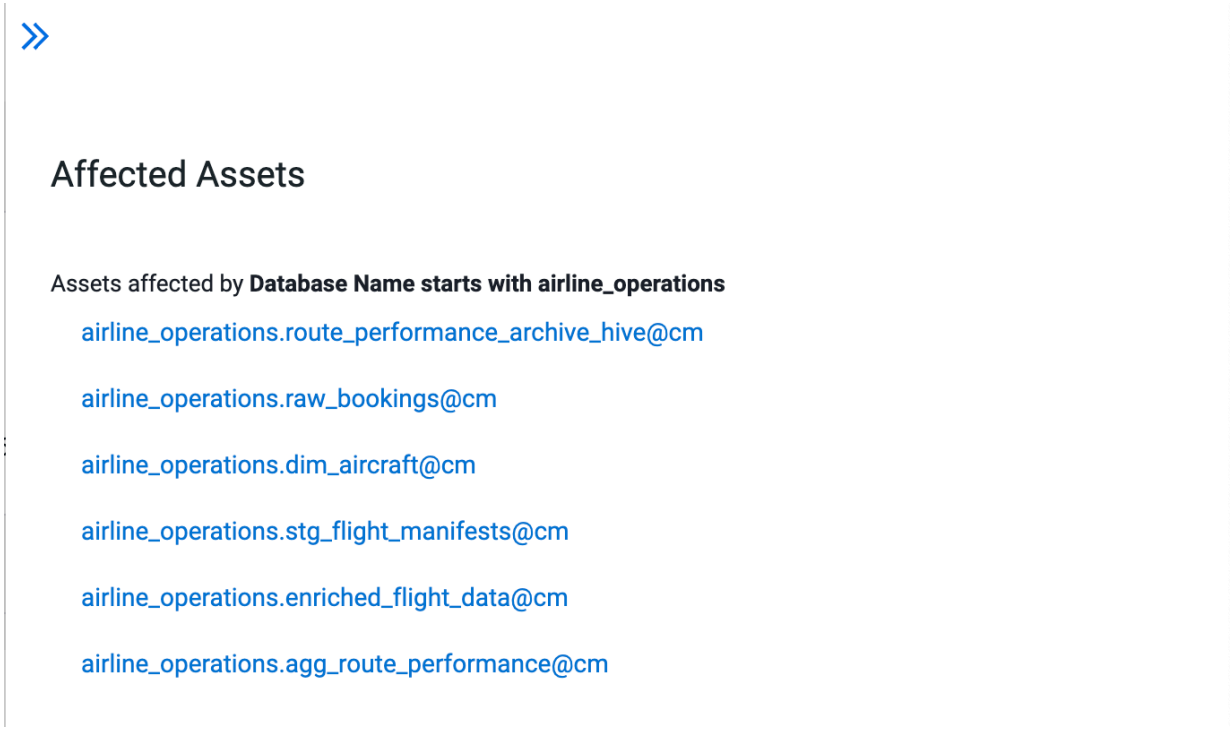
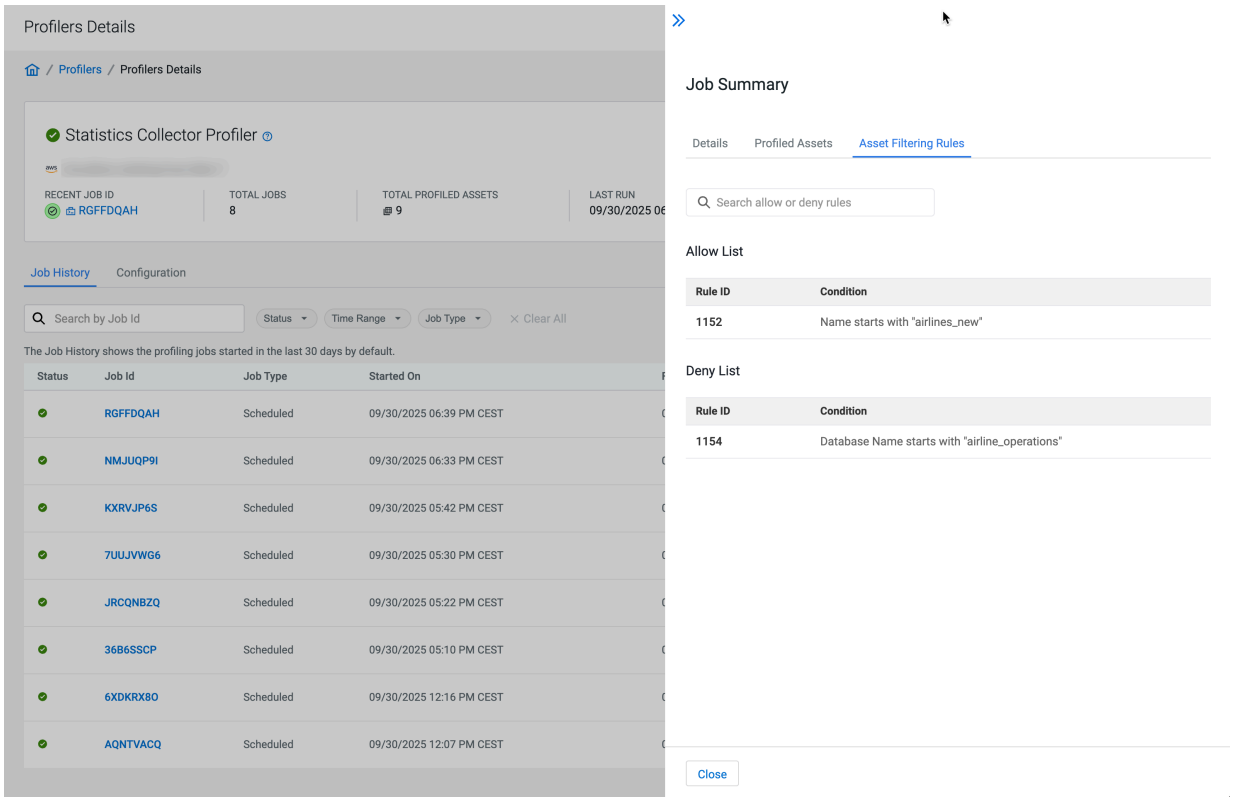


Figure 6: Affected Assets in Asset Filtering Rules configuration

[3] By Creation Date, Greater than 7 days means an asset older than seven days. Less than 7 days means an asset younger than seven days.



Job Summary shows the asset filtering rules applied for the particular profiling job:



Profiler tag rules in Compute Cluster enabled environments

You can use preconfigured tag rules or create new rules based on regular expressions and values in your data to be profiled by the Data Compliance. When a tag rule is matching your data, the selected Apache Atlas classification (also known as a Cloudera Data Catalog tag) is applied.

Note: The improved tag rules are available for Compute Cluster enabled environments. In VM-based environments, tag rules are valid for all data lakes, while tag rules in Compute Cluster enabled environments are data lake specific.

Tag rule types

Tag Rules are categorized based on their type into the following groups:

- **System Defined**: These are built-in rules that cannot be edited. You can only enable or disable them for your data.

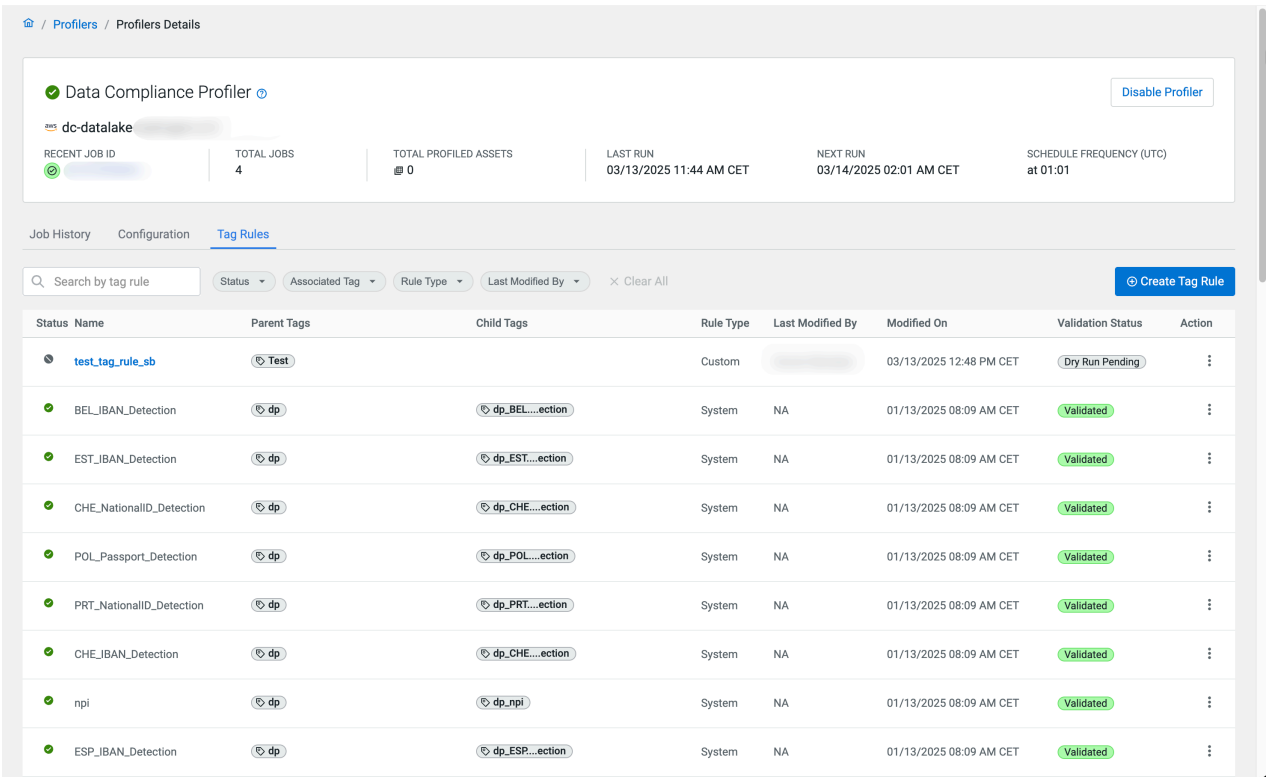
  Note:

  Calculation for System Defined tag rules:

  The match threshold is set to 70% for column values with the given regex. The column value matching is given a weightage of 85% in the final score and the remaining 15% is associated with the column name matching.
- **Custom**: Tag rules that you create, edit and deploy on clusters after validation will appear under this category.

  Click the ⋮ icon in the **Action** column to enable your custom tag rules. You can also edit these tag rules.

Profilers Details



After creating your rule, you have to validate them with test data by completing a Dry Run and, only then you can click Enable.

Note: Tag Rules can be temporarily suspended.

Tag rule inputs

Tag Rules can be applied based on the following inputs:

Input type	VM based environments	Compute Cluster enabled environments
Column name value	Manually entered regex pattern	• Manually entered regex pattern • Uploaded regex pattern
Column value	Manually entered regex pattern	• Manually entered regex pattern • Uploaded regex pattern • CSV files with data which will be matched against column values for your tables in your data lake.
Table name		• Manually entered regex pattern • Uploaded regex pattern

Match thresholds and weightage

In Compute Cluster enable environments, you can adjust the **Column Value Weightage** for tag rules defined with regex patterns. The column value weightage percentage complements the column name weightage to 100%. This means that if you set the column value weightage to 80%, the column name adds to the final match score either 20 or zero, The reason for this is that column name matching can have only binary results (match or no match), while column value match is the number of matching values (rows) from all values in the column.

The System Deployed rules have a preset match threshold: A matching column name means a 15% confidence value. This is increased by 85% by a matching column value.

Tag rule testing

After creating your tag rule, you have to test it:

By Compute Cluster enabled environments, review them with data uploaded in a file, then save them to reach the Dry Run Pending status. Tag rules in this status must be also tested with a Dry Run on a subset of your data (up to 10 tables) in the data lake before deploying them. A Dry Run is a special on-demand profiling job.

Tag handling by tag rules

Successfully tested and enabled tag rules apply Atlas classifications or synchronized Cloudera Data Catalog tags to tables, columns.

In Compute cluster enabled environments, the parent-child tag relationships are respected. When the column value matches a child tag, the table receives the parent tag.

Note:

Tags created in Cloudera Data Catalog automatically receive a status attribute. This is can be used to identify the association of the tag with the asset.

Creating tag rules in compute cluster environments

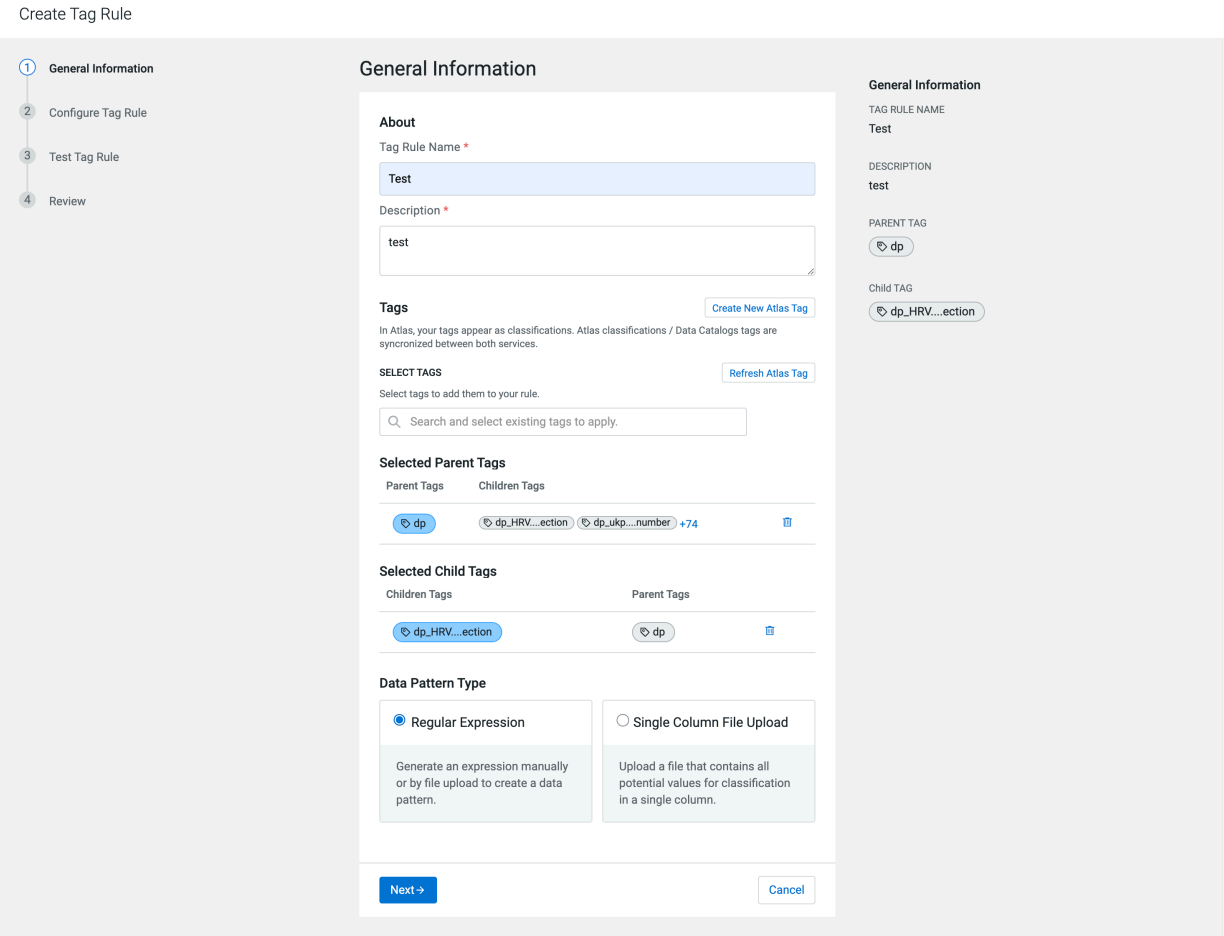
With tag rules, you can apply Apache Atlas classifications to your assets based on regex expressions or similarity to a set of values in a table.

About this task

Procedure

1. To start applying tags, go to **Profilers** and select your data lake.

2. Go to Profilers Data Compliance Tag Rules .
3. Click + Create Tag Rule.
4. Name your tag rule and add a description to it in **General Information**.



5. Select the tags to be applied from the list of available tags synchronized from the list of Atlas classifications.

   If you select a child tag, its parent tag is also automatically selected. By default, if the child tag is applied to a column, the table receives the parent tag.
6. Select your **Data Pattern Type**:

Option	
Regular Expression	You can upload a text file containing your regex expression or directly type it in the **Configure Tag Rule** page. The required format of the CSV file can be seen by clicking Download Sample Tag Rule. Continue in step 7 on page 27.
Single Column File Upload	Upload a CSV file with values to be matched against the actual values in your tables. After uploading your file, continue with step 11 on page 28.

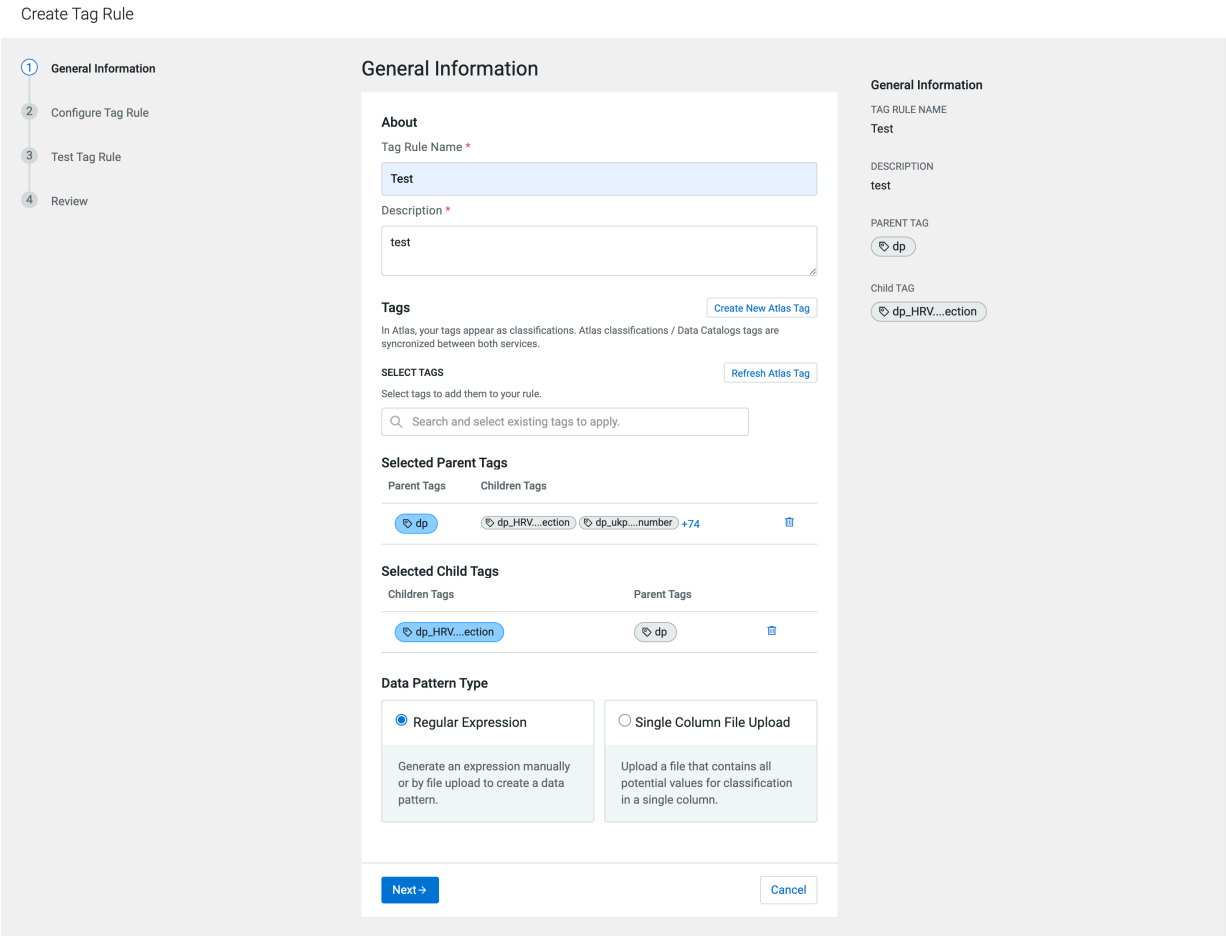
Creating regular expression based tag rule:

7. Define your regular expression for the table name.

Note: Cloudera recommends using PCRE2 compatible regular expressions. Non-compliant regular expressions may show reduced performance.

For more information, see PCRE - Perl Compatible Regular Expressions.

8. When using **Column Level** regex expressions, you can define multiple expression for both of the following:
   - Column Name
   - Column Values



Note: Regular expressions matching the same type of entity (column name or value) have the OR logical relationship between them. When using multiple regular expressions of the same type (table name, column name or value), even if one of the regular expressions match, it is considered as a match.

9. Define the Column Value Weightage in percentage with the slider.

   The remainder percentage is the column name weightage percentage. The results of the individual regex matches are weighted according to this setting before determining the final result confidence for applying the tag.

Note: A correctly formatted file is automatically processed by Cloudera Data Catalog. All details will be filled in this case.

Tag rule testing:

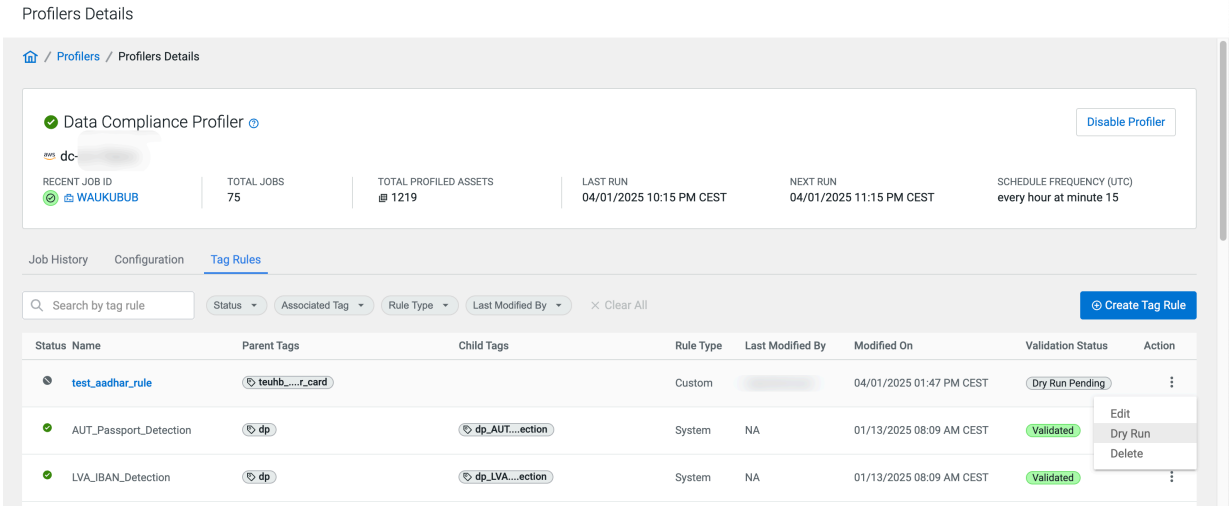
10. You can make a sanity check of your tag rule in **Test Tag Rule** by uploading a sample dataset in CSV format.

Note: A final test called "Dry Run" is still needed to be passed to enable your tag rule.

11. Review all your input before clicking Create Tag Rule.
    a) Click Confirm to finalize your tag rule.

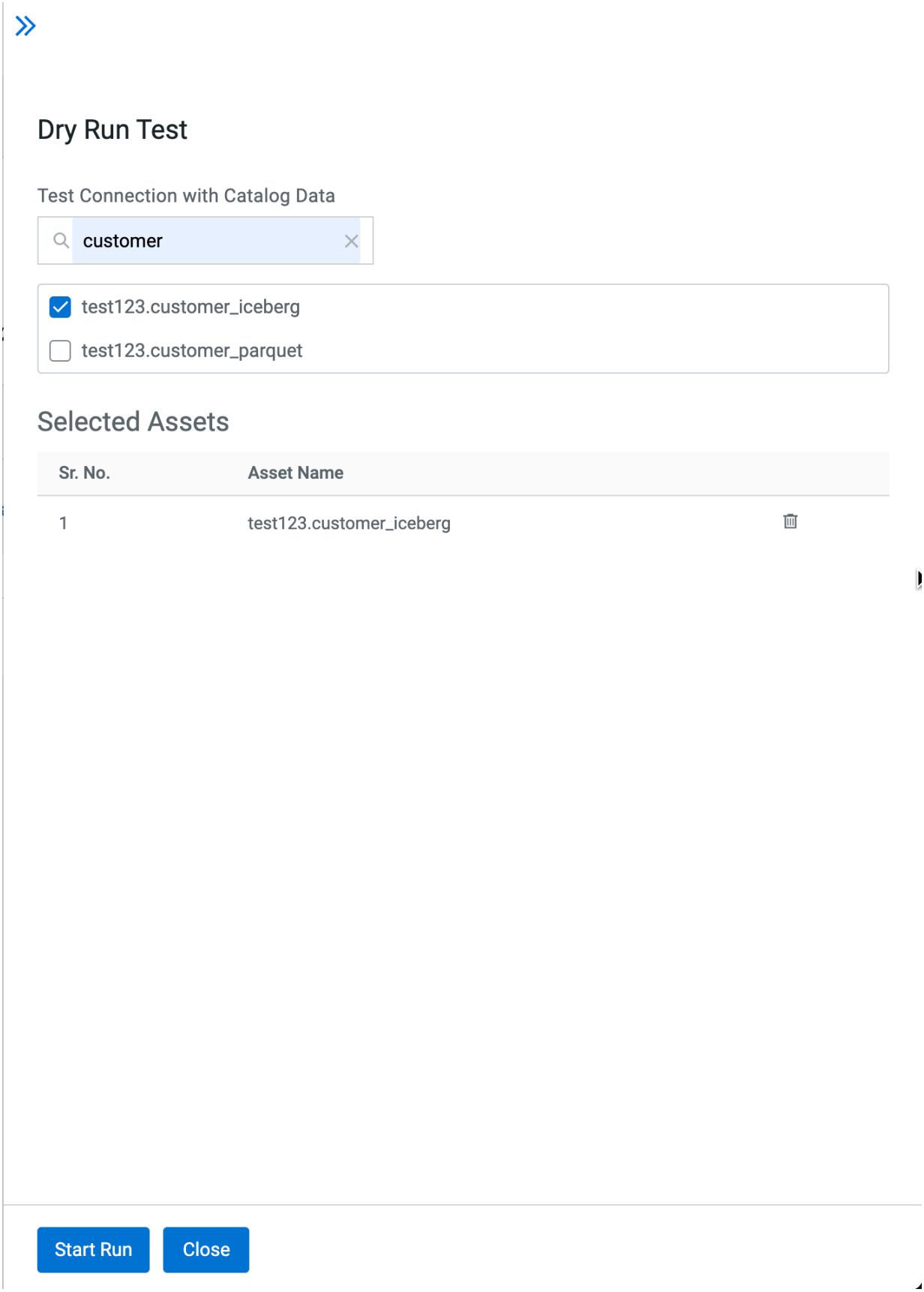
    Your tag rule is created with **Status** Disabled( ) and the **Test Status** will be Test Pending.

12. Click ⋮ > Dry Run.



The **Dry Run Test** pane opens.

13. Click Run to start an on-demand dry run profiling job on up to 10 tables from your data.



Your tag rule becomes VALIDATED after a successful dry run.

14. After the "Dry run" test was passed, click ⋮ > Enable to start your using your tag rule on your live data.

Approving Data Compliance Profiler tags

You are able to check and approve tags created by the Data Compliance Profiler assigned to your assets before applying them.

About this task

Once your Data Compliance Profiler job completes, you can check your profiled assets one-by-one and decide to keep the suggested tags or remove them from asset before they are synchronized to Apache Atlas.

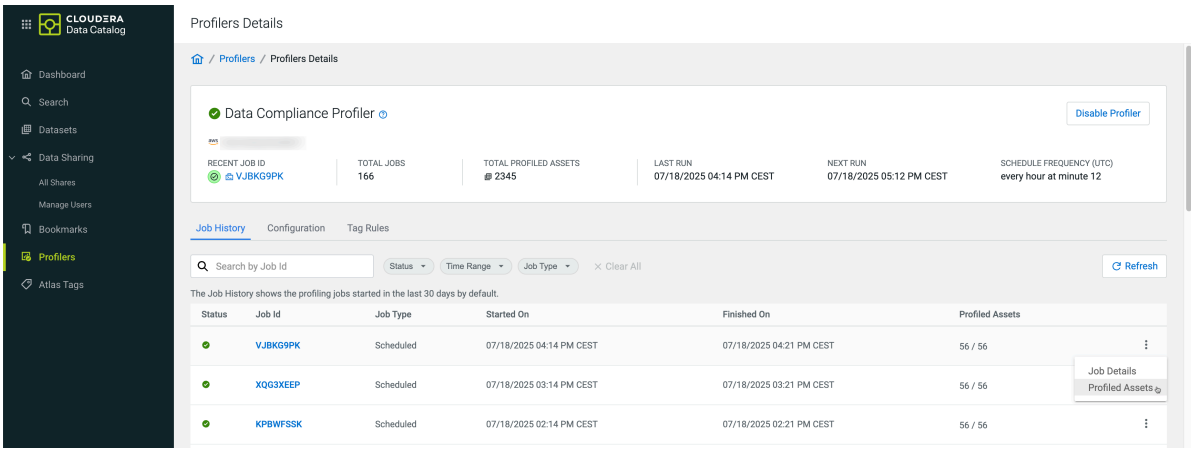
This gives you the ability, for example, to correct the tagging of assets mistakenly marked as PII. Incorrectly applied tags can affect your tag-based policies, unexpectedly changing the access of your users.

Before you begin

Note: This feature is only available for the Data Compliance Profiler in the Compute Cluster enabled environment.

Procedure

1. Once your profiler job is completed, go to Profilers Data Compliance Profiler Job History Profiled Assets .
   a) Select the row with the relevant *JOB ID* and click ⋮, then open **Profiled Assets**.



2. Check the content of either of the following columns:

Option	
Asset Name	Opens the **Asset Details**.

Option	
Suggested Tags Count	Opens the list of identified tags for the asset. If the asset has columns, you can see the tags assigned to each of them.

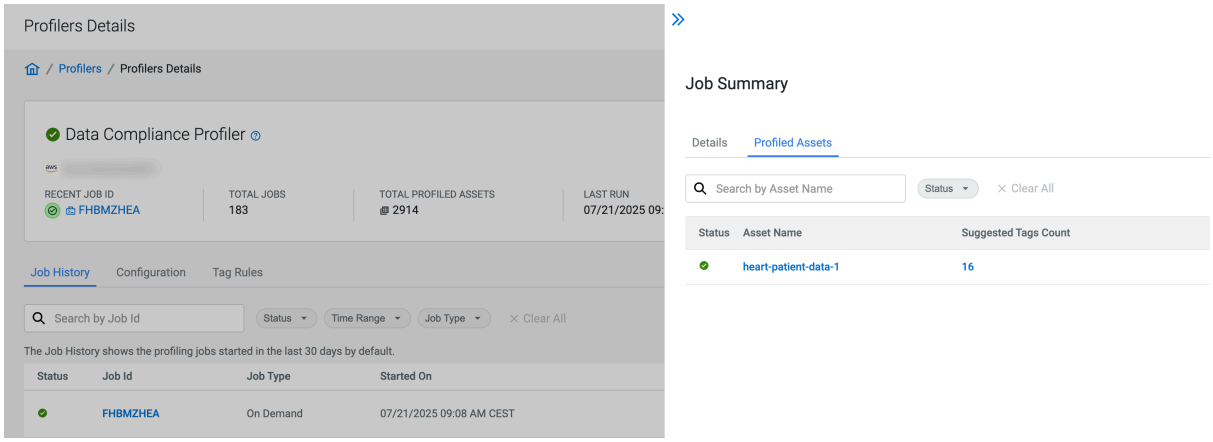
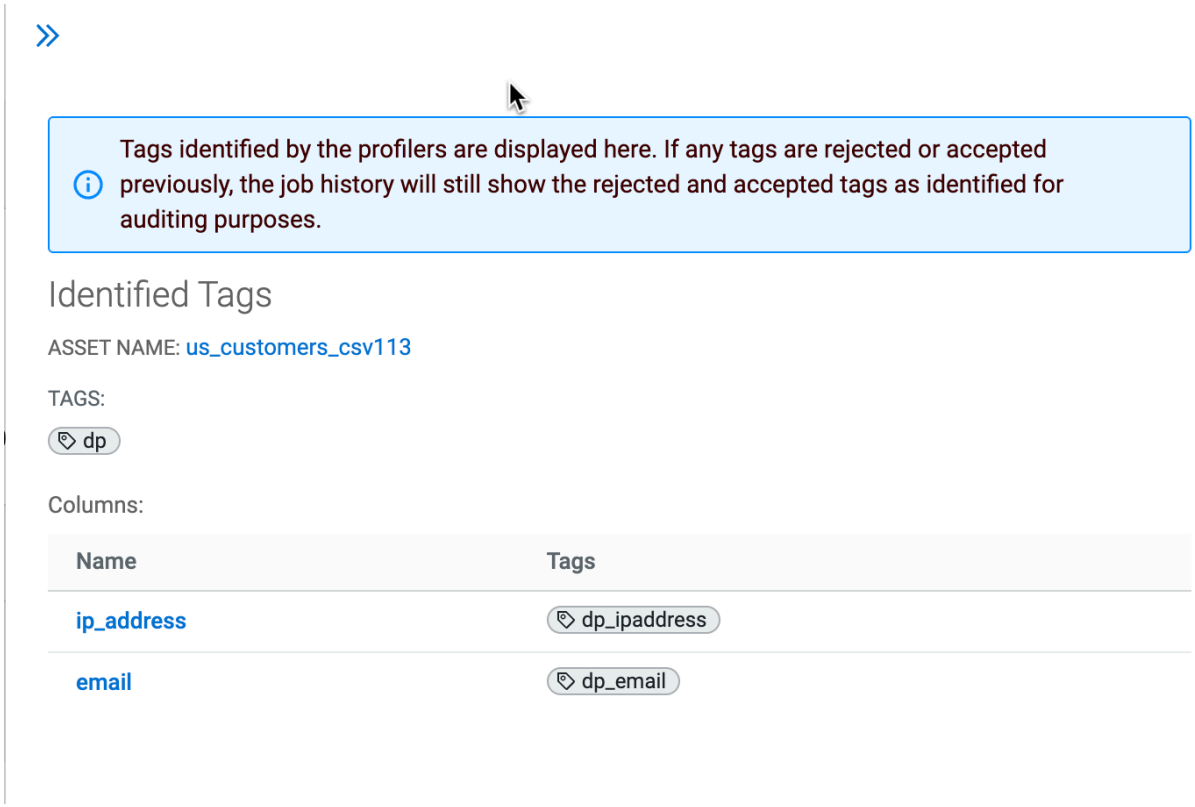


Figure 7: List of tags in Suggested Tags Count

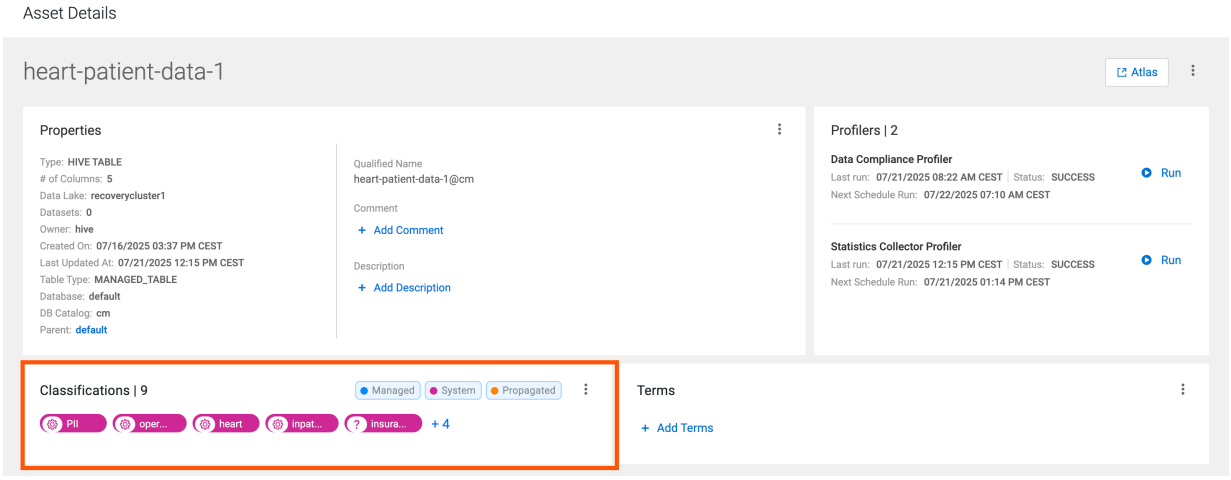


3. Click the asset whose suggested tags you want to review.

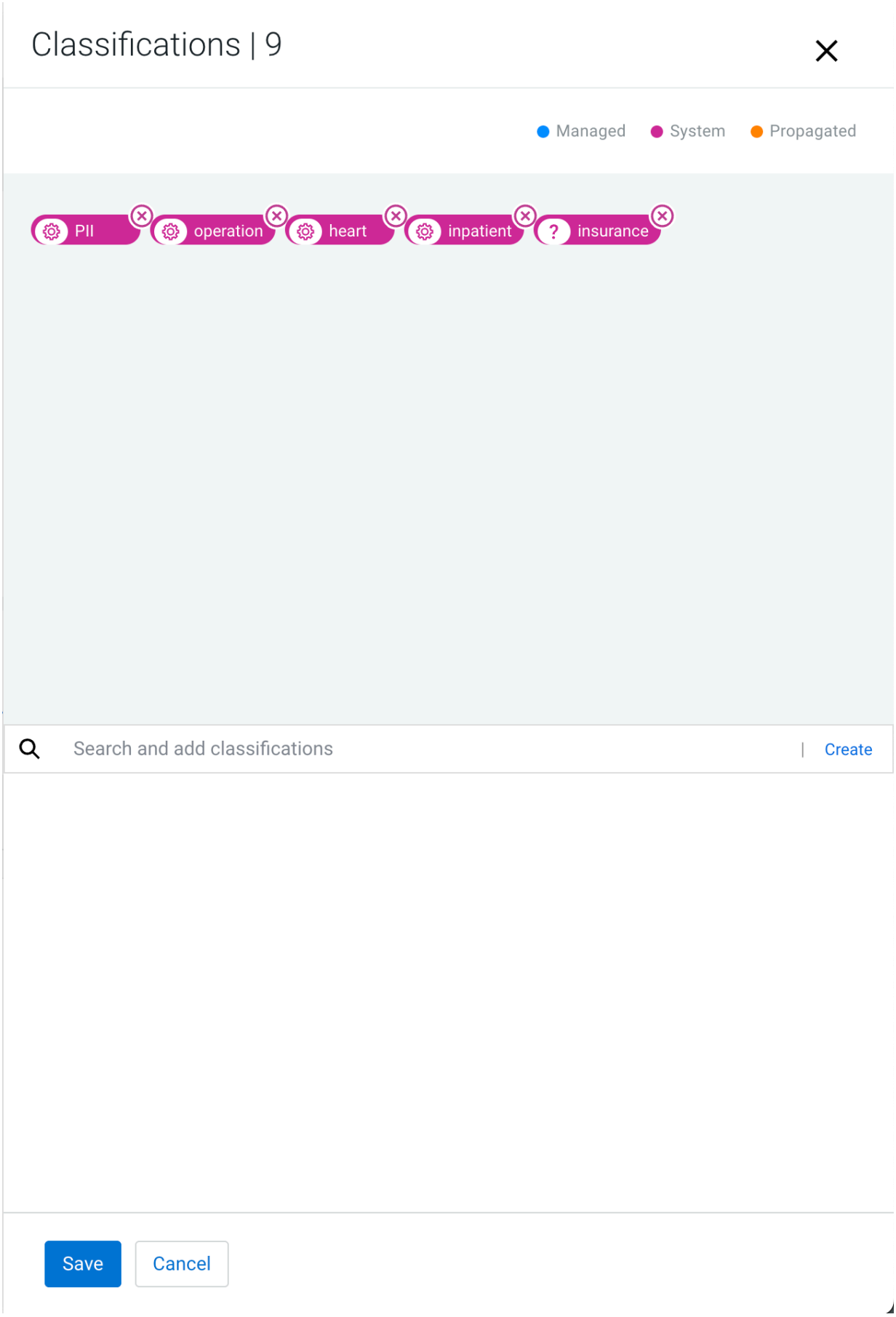
4. Click Asset Details Classifications ⋮ Edit .

   Tags (Apache Atlas classifications) pending approval are marked with the ? icon.

   Approved tags are marked with the ⚙ icon.



   You can remove suggested and already applied tags by clicking the ⊗ icon.


Classifications | 9
Managed
System
Propagated
PII
operation
heart
inpatient
insurance
Search and add classifications
Create
Save
Cancel

5. Click Save to apply your changes.

Configuring the Statistics Collector profiler

You can configure the scheduling and the available resources for your profiler.

Procedure

1. Go to **Profilers** and select your data lake.
2. Select a schedule to run profiler using either UNIX Cron Expression or the Basic scheduler

Note: Both the Basic and Cron Expression scheduler (Unix in Compute Cluster enabled environments cron jobs) use the UTC timezone instead of the local timezone of the user.

Figure 8: Profiler schedule with cron expression

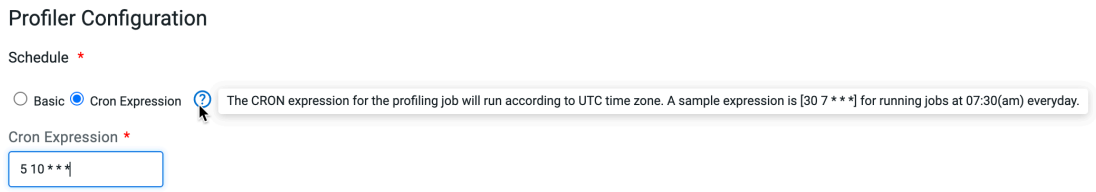
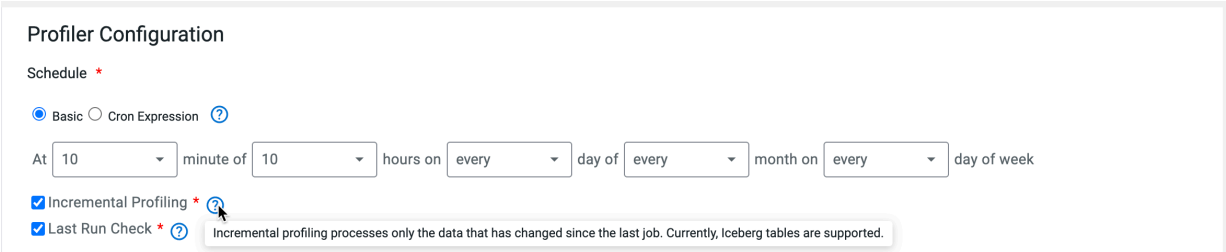


Figure 9: Profiler schedule with natural language



3. Select Incremental Profiling when needed.

   Using Incremental Profiling can decrease the compute resources and the time needed for the profiling job by processing only the information (only Iceberg tables) updated or added since previous job.

   Using Incremental Profiling, you can refine the results from the Last Run Check. Incremental Profiling checks the data (rows) in assets, while Last Run Check filters complete assets.

   Note: By Statistics Collector Profilers, the profiler compares the aggregated metrics between old and newly added data. Depending on the differences, this can slightly skew results. It is highly recommended to process the complete dataset time to time for the most accurate results.
4. Select Last Run Check and set a period in Day Range if needed.

   Note:

   The Last Run Check enables profilers to avoid profiling the same asset on each scheduled run.

   If you have scheduled a cron job, for example set to start in about an hour, and have enabled the Last Run Check configuration for two days, this setup ensures that the job scheduler filters out any asset which was already profiled in the last two days.

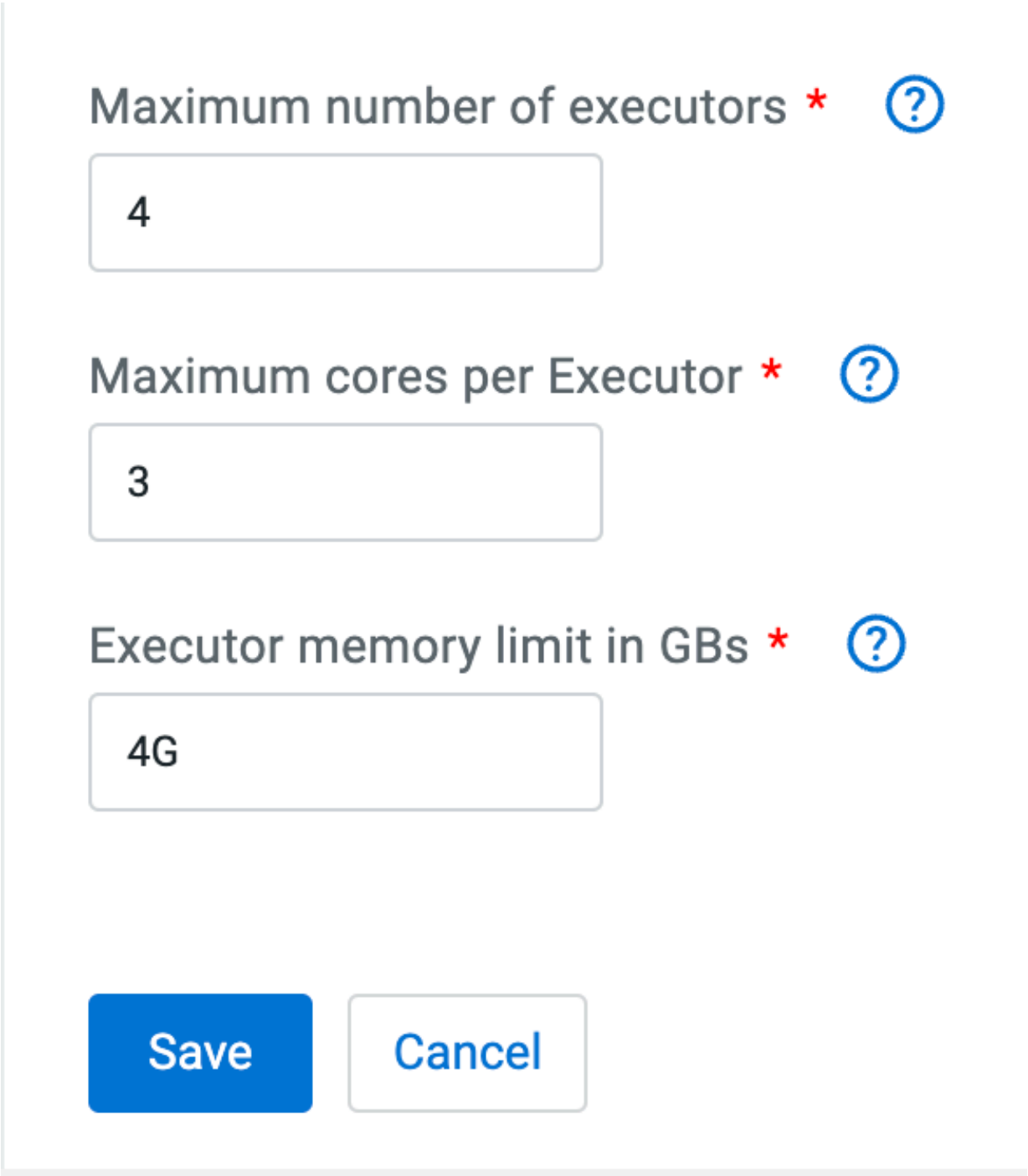
   If the Last Run Check configuration is disabled, assets will be picked up for profiling as per the job cron schedule, honoring the asset filter rules.

   The Last Run Check precedes Incremental Profiling.

5. Continue with resource settings:
   a) Set the Maximum number of executors

      Indicates the number of workers that are used by the distributed computing framework. The recommended value is at least 10 executors.
   b) Set the Maximum cores per executor

      Indicates the maximum number of cores that can be allocated to an executor.
   c) Set the Executor memory limit in GBs



6. Click Save to apply the configuration changes to the selected profiler.

7. Add **Asset Filtering Rules** as needed to customize the selection and deselection of assets which the profiler profiles.

   Note:

   - Profiler configurations apply to both scheduled and on-demand profiler jobs.
   - Asset filtering rules apply to assets, such as tables, and not to complete databases.
   - Multiple asset filtering rules are evaluated together as if connected by the OR operator.
   - In Compute Cluster environments, you cannot enable conflicting Allow and Deny list rules at the same time. Enabling conflicting rules results in an error message.

   a) Set your **Deny List** and **Allow-list**.

      The profiler will skip profiling assets that meet any criteria in the **Deny List** and will include assets that meet any criteria in the **Allow List**.

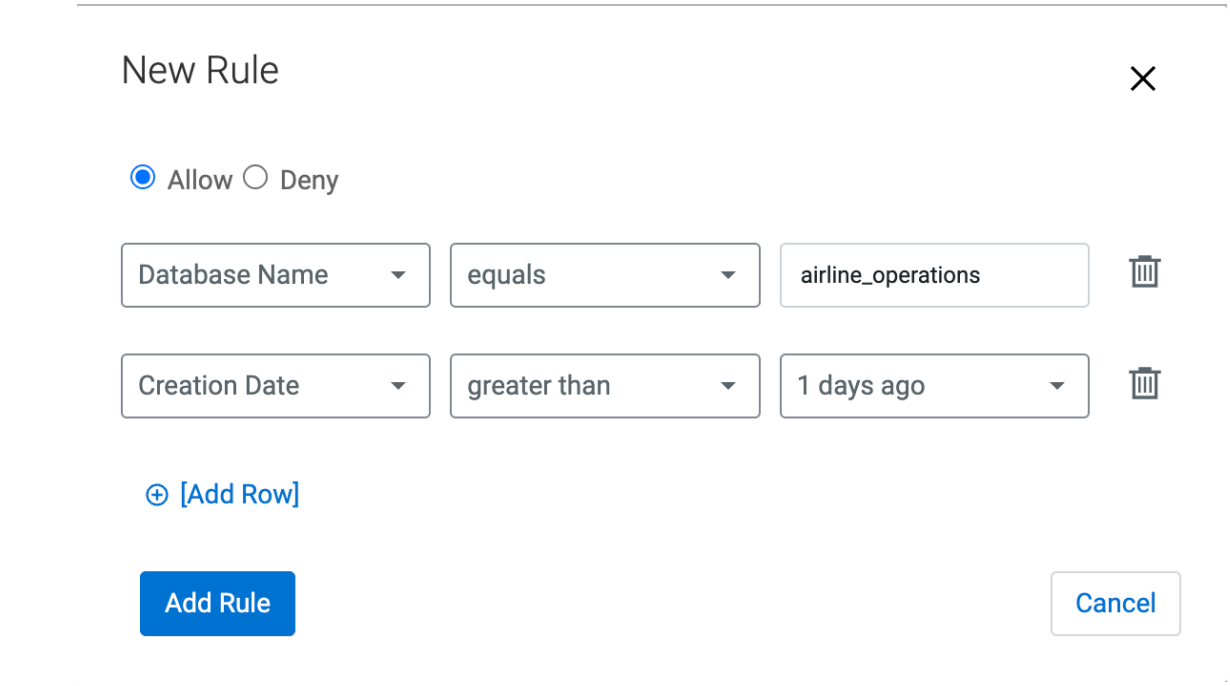
      1. Click Add New Rule to define new rules.
      2. Use the radio buttons to define your new rule for the Allow or Deny List.
      3. Select the key from the drop-down list and the relevant operator. You can select from the following:

Key	Operator
Database name	• equals • starts with • ends with
Name (of asset)	• equals • contains • starts with • ends with
Owner (of asset)	

Key	Operator
Creation date[4]	• greater than • less than

Note: **Name** refers here to the actual name of the asset and not to its **Qualified Name**.

4. Enter the value corresponding to the key. For example, you can enter a string as mentioned in the previous example.
5. Click Add Rule. Once a rule is added (enabled by default), you can toggle the state of the new rule to enable it or disable it as needed.



Note: You can check the list of assets impacted by your rule by clicking ⋮ > Affected Assets.

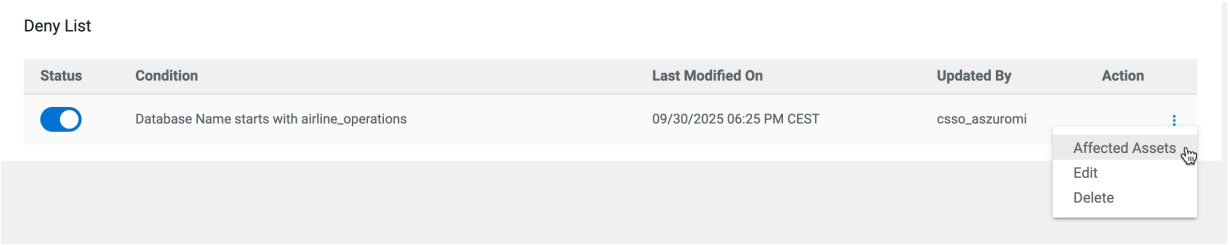
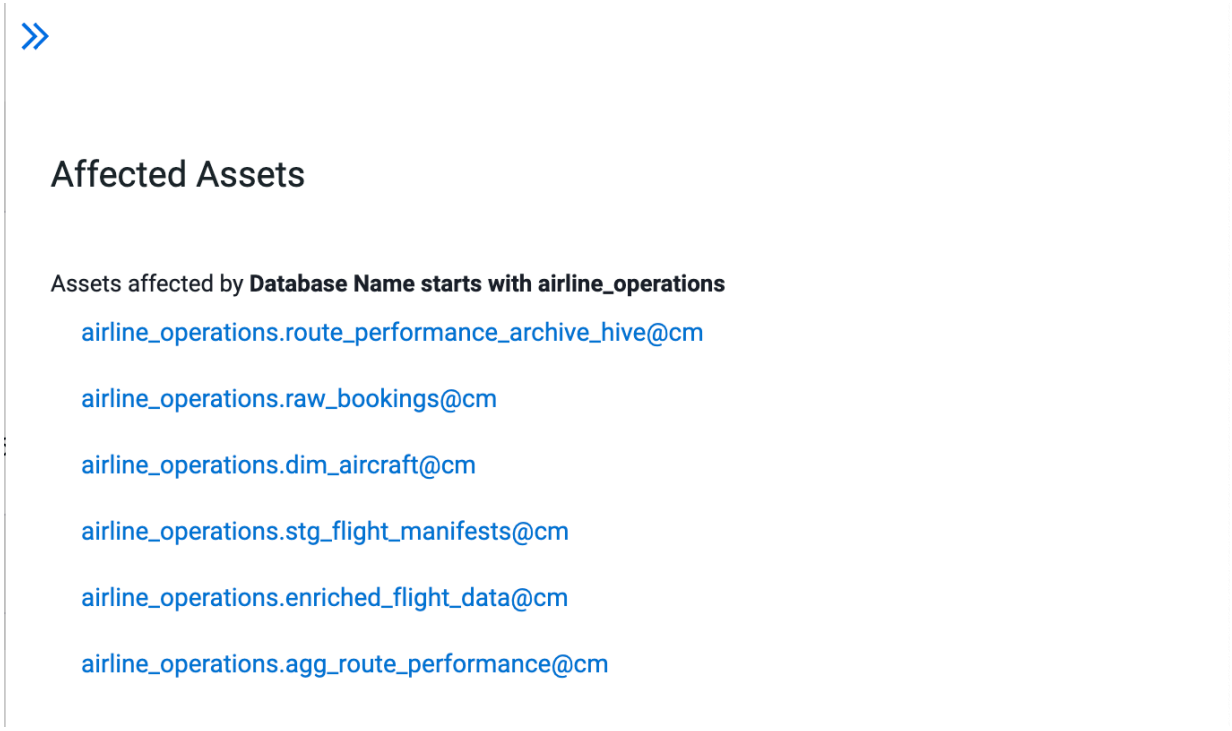
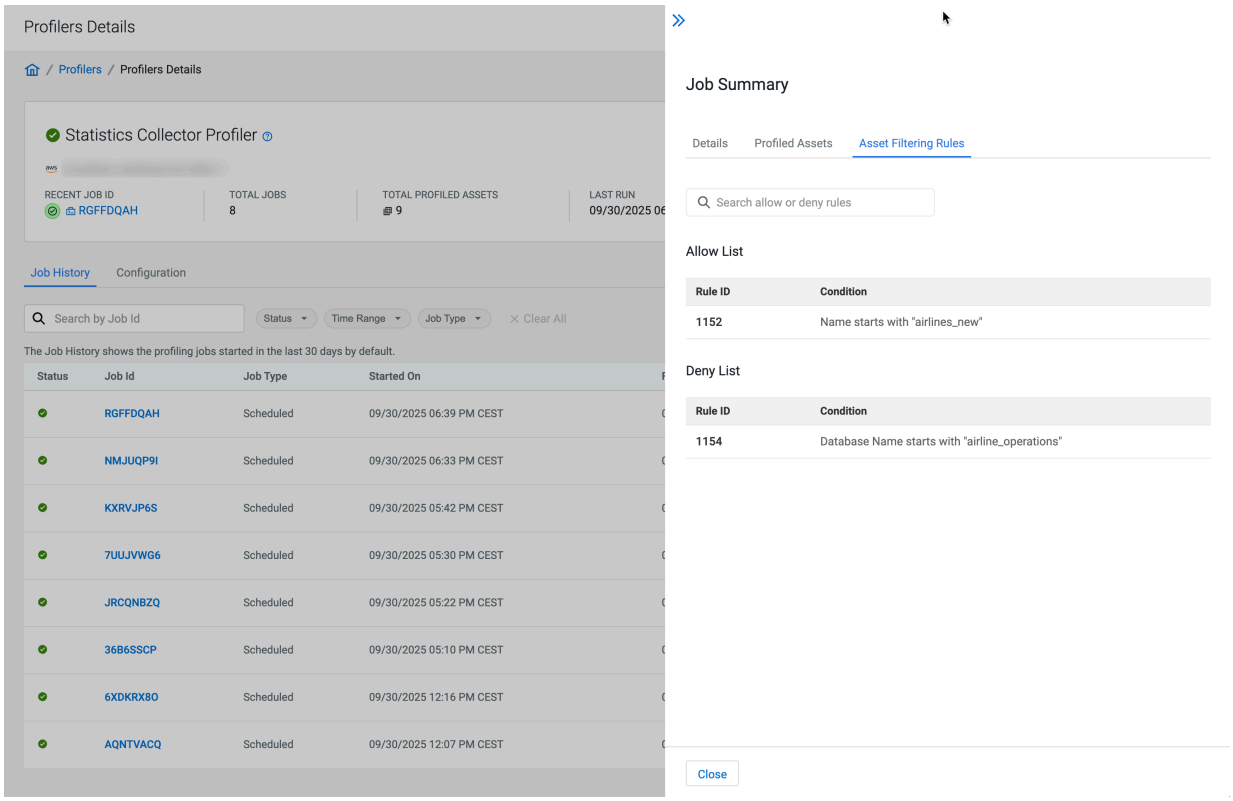


Figure 10: Affected Assets in Asset Filtering Rules configuration

[4] By Creation Date, Greater than 7 days means an asset older than seven days. Less than 7 days means an asset younger than seven days.



Job Summary shows the asset filtering rules applied for the particular profiling job:



Understanding the Cron Expression generator

In Profilers > Profilers Details > Configuration > All Configurations, a cron expression can be used to define when the profiler schedule executes and visualizes the next execution dates of your profiling jobs.

The Unix (in Compute Cluster enabled environments) and quartz (in VM-based environments) cron expressions use the following typical format:

Each * in the cron represents a unique value.

For VM-based environments

Schedule: * * * * * ? *

In this format the * characters represent the following units:

1. seconds (0-59)
2. minutes (0-59)
3. hours (0-23)
4. day of the month (1-31)
5. month (1-12 or JAN-DEC)
6. day of the week (1-7 or SUN-SAT)
7. year (optional, 1970-2099)

Consider the following examples:

1 2 3 2 5 ? 2021

This cron expression is scheduled to run the profiler job at: 03:02:01am, on the 2nd day, in May, in 2021.

Note: The ? character is a replacement for the day of the week (or a day of the month). It is not specified on which day (or month) of the week the job has to run.

*** * * ? * ***

Every second

0 * * ? * *

Every minute (every 60th second)

0 0 * ? * *

Every hour (every 60th minute)

0 0 13 * * ?

At 13:00:00 every day

0 0 13 ? * WED

At 13:00:00, on every Wednesday, every month

0 0 13 ? * MON-FRI

At 13:00:00, on every weekday, every month

0 0 12 2 * ?

Every month on the 2nd, at noon

For Compute Cluster enabled environments

Cron Expression: 0 18 * * *

In this format the * characters represent the following units:

1. minute (0-59)
2. hour (0-23)
3. day of the month (1-31)
4. month (1-12)
5. day of the week (0-6)

Consider the following examples:

30 10 15 5 *

This cron expression is scheduled to run the profiler job at: “At 10:30 on 15th day-of-month in Ma y.”

Note: The * character is a replacement for the day of the week. It is not specified on which day of the week the job has to run.

30 10 * 5 7

This cron expression is scheduled to run the profiler job at: “At 10:30 on Sunday in May”.

Note: The * character is a replacement for the day of the month. It is not specified on which day of the month the job has to run.

5 * * * *

Every fifth minute of the hour. (18:05, 19:05, 20:05, etc.)

5 5 * * *

At 05:05 every day.

5 5 5 * *

At 05:05 on every fifth day of the month. (07-05 05:05:00, 08-05 05:05:00, 09-05 05:05:00, etc.)

5 5 5 5 *

At 05:05 on every fifth day of May (fifth month of the year); (2026-05-05 05:05:00, 2027-05-05 05:05:00, 2028-05-05 05:05:00, etc.)

5 5 5 5 5

At 05:05 on fifth day of the month OR if its a Friday (fifth day of the week) in May (fifth month of the year); 2026-05-01 05:05:00, 2026-05-05 05:05:00, 2026-05-08 05:05:00, etc.

You can change the value of cron as and when it is required depending on how you want to schedule your profiler job.

Note: Both Unix (in Compute Cluster enabled environments) and quartz CRON jobs (in VM-based environments) use the UTC timezone instead of the local timezone of the user.

On-Demand Profilers in compute cluster environments

You can use On-Demand Profilers to profile specific assets without depending on the cron-based scheduling of profilers jobs. The On-Demand Profiler option is available in the Asset Details of the selected asset.

The following image shows the **Asset Details** page of an asset. You can run an On-Demand Profiler for Hive Column Profiler and Cluster Sensitivity Profiler by clicking on the appropriate Run button. The next scheduled run provides details about the next scheduled profiling for the respective profilers.

Note: You can use the On-Demand Profiler feature to profile both external and managed tables.

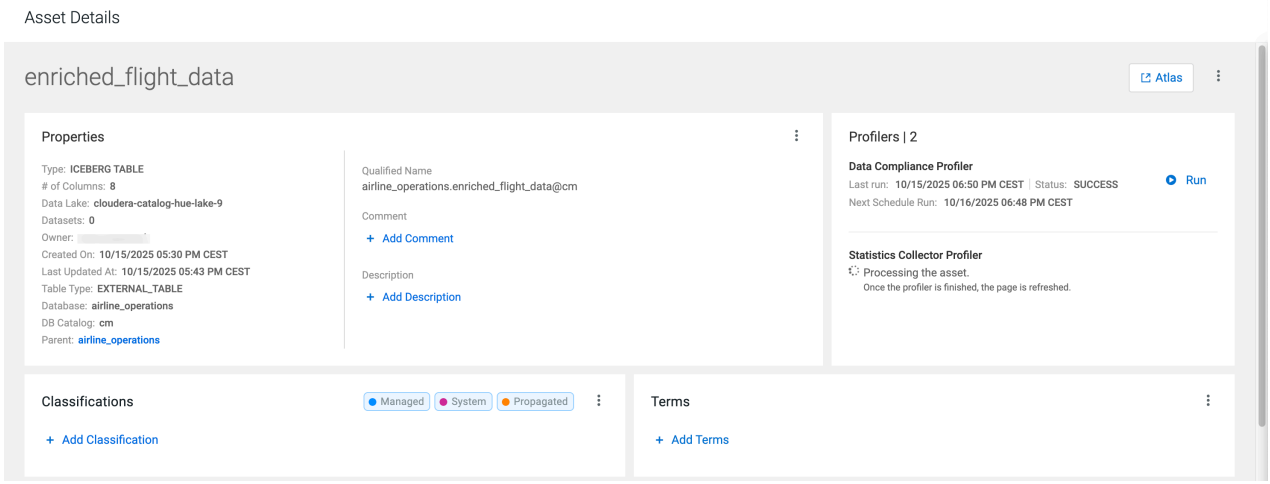
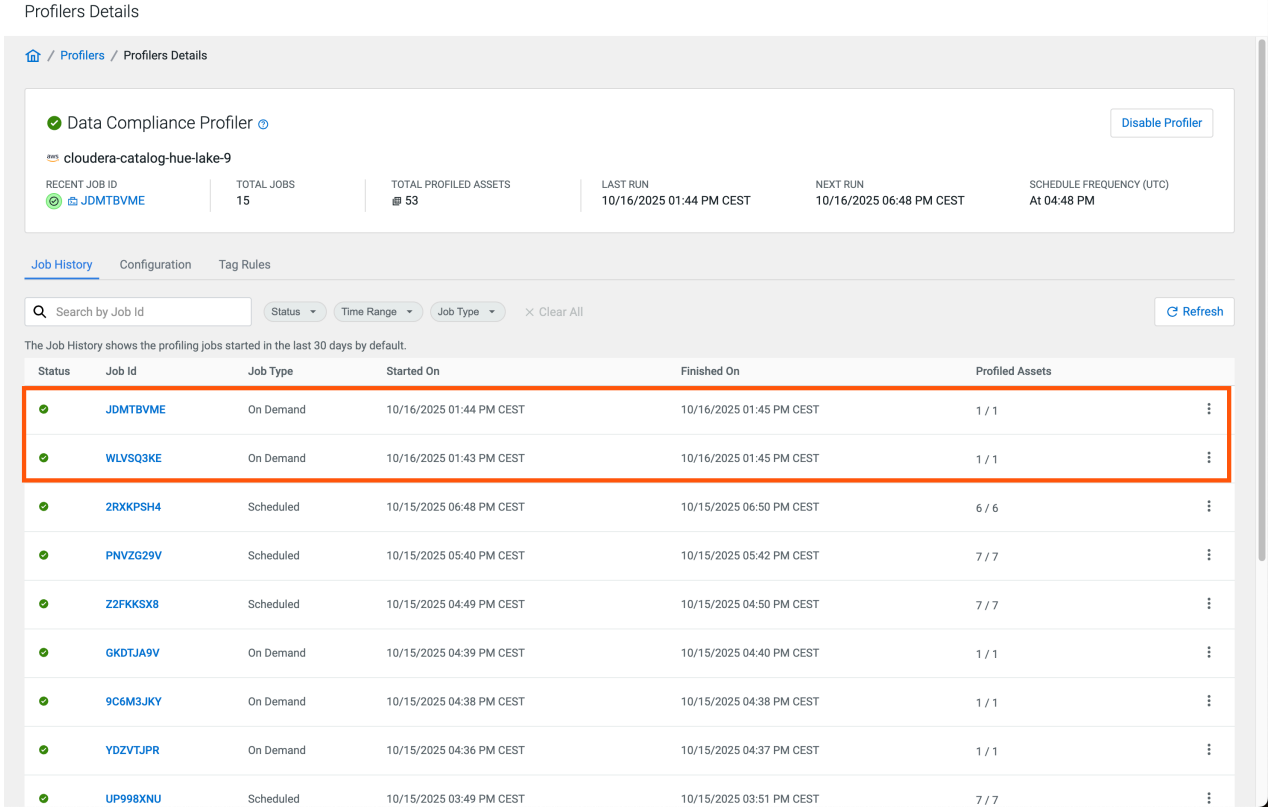


Figure 11: Tracking on-demand profilers



Note:

- In Compute Cluster-enabled environments, Iceberg tables can be also profiled with the On-Demand Profiler.
- Profiler configurations apply to both scheduled and on-demand profiler jobs.

Deleting profilers in Compute cluster enabled environments

In Compute Cluster enabled environments deleting the profiler jobs removes all the Data Compliance profiler rules and other updates to the specific cluster. It could also cause loss of data specific to currently applied rules on the deleted profiler cluster.

About this task

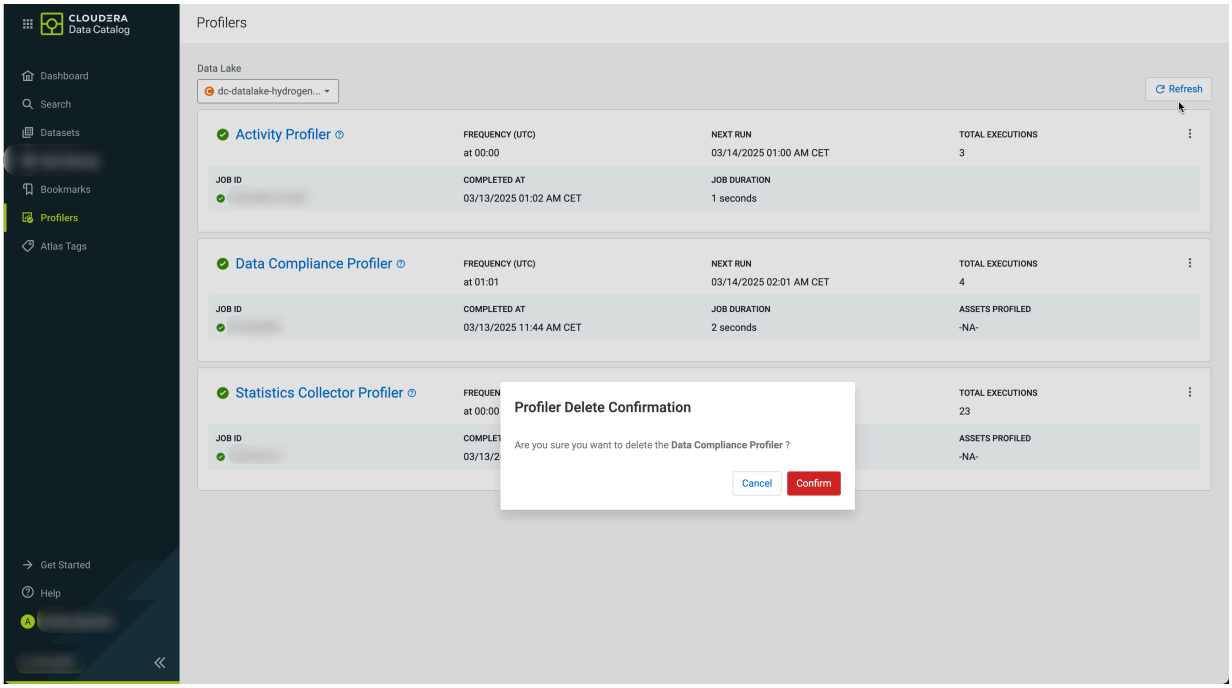
Note:

- In a Compute Cluster enabled environment, when you delete the scheduled jobs, the associated Kubernetes cron job object is deleted from the Kubernetes cluster.
- The associated data of the profilers from the Cloudera Management Console database is also deleted for the specified data lake.

Procedure

1. On the **Profilers** page, select the data lake from the drop-down.
2. Click Delete Profiler in the action menu (⋮) for the profiler you want to delete.
3. Confirm the deletion in the message dialog box.

Figure 12: Deleting a profiler in a Compute Cluster enabled environment



4. Click Confirm and repeat the step for each profiler.

Note: It might take a couple of minutes until all profilers are deleted as a running profiler cannot be stopped. Periodically click Refresh to update the status.

Note: You cannot delete a profiler while it is running.

Note: By deleting the last profiler, you also delete the namespace and underlying infrastructure associated with the profilers.

The profiler cluster is deleted successfully.