Cloudera Runtime 7.2.16

Using Hue

Date published: 2020-07-28
Date modified: 2023-01-11

https://docs.cloudera.com/

Legal Notice

© Cloudera Inc. 2025. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

About using Hue 5

Accessing and using Hue 5

Viewing Hive query history 11

Viewing Hive query information 11

Viewing explain plan for a Hive query 12

Viewing Hive query timeline 13

Viewing configurations for a Hive query 14

Viewing DAG information for a Hive query 15

Comparing Hive queries in Hue 19

Terminating Hive queries 20

Using SQL to query HBase from Hue 20

Querying existing HBase tables 21

Enabling the SQL editor autocompleter 22

Rerunning a query from the Job Browser page 23

Using governance-based data discovery 23

Searching metadata tags 23

Using Amazon S3 with Hue 24

Enabling S3 browser for Hue configured with IDBroker 24

Enabling S3 browser for Hue configured without IDBroker....25
Populating an S3 bucket....26
Creating a table from an Amazon S3 file....27
Exporting query results to Amazon S3....27

Using Azure Data Lake Storage Gen2 with Hue....29
Enabling ABFS file browser for Hue configured with IDBroker....29
Enabling ABFS file browser for Hue configured without IDBroker....31

Granting permission to access S3 and ABFS File Browser in Hue....31

Supported non-ASCII and special characters in Hue....32

Options to rerun Oozie workflows in Hue....34

Options to restart the Hue service....35

About using Hue

Hue provides a one-stop querying experience in Cloudera Data Platform (CDP) that leverages Hive and Impala SQL queries.

Accessing and using Hue

Get started using Hue by analyzing and visualizing your data with Impala and Hive SQL query engines.

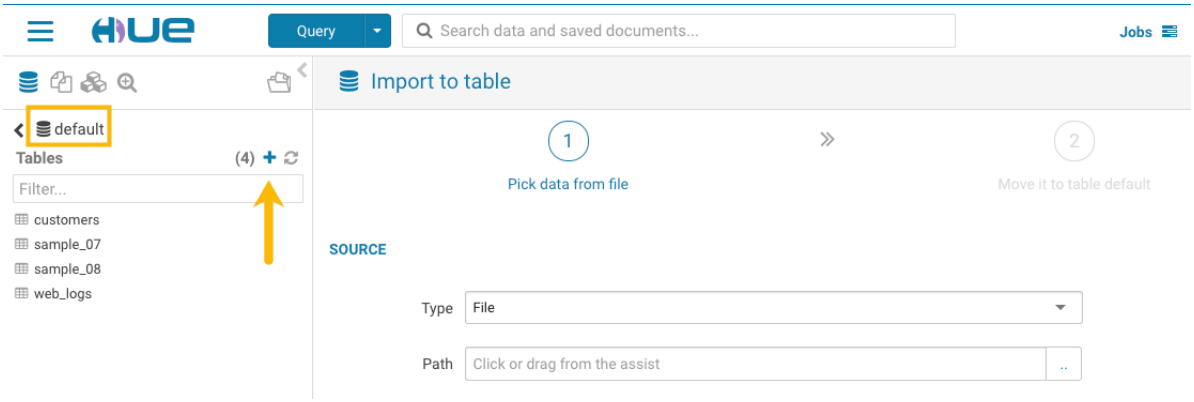
About this task

To try Hue without having an account, try running sample queries on http://demo.gethue.com/.

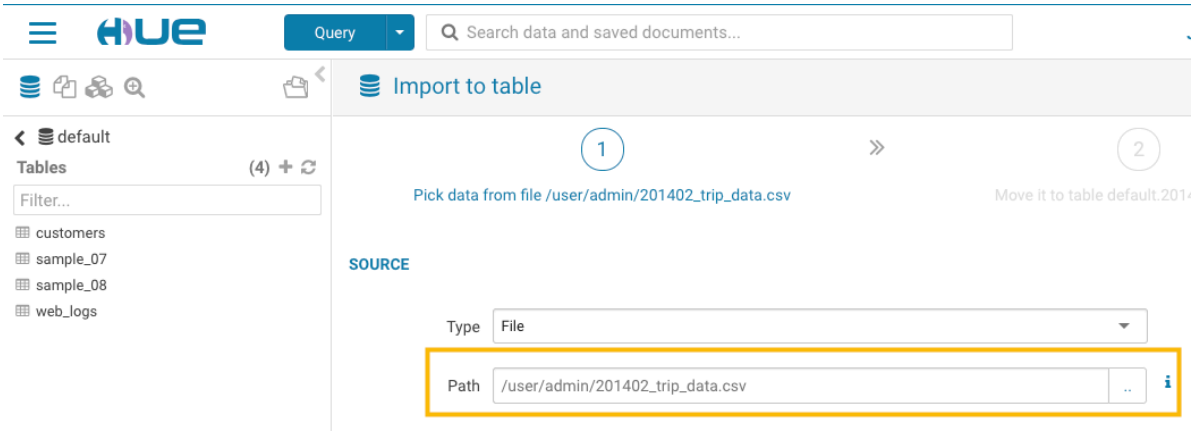
Procedure

1. Download and unzip one year of bike trips from the Bay Area Bike Share program. This file is about 80 MB in size.

2. Create a table from the ~/babs_open_data_year_1/201402_babs_open_data/201402_trip_data.csv file found in the unzipped babs_open_data_year_1.zip file:
   a) In the Cloudera Manager, select Hue WebUI Hue Load Balanced to launch Hue.
   b) In the left navigation panel of Hue, make sure the default database is selected, and click the plus sign to create a table as shown in the following image:



   If the default database is not selected, click the "less than" icon that is next to the database icon in the left panel. This enables you to select the default database.
   c) In the center panel Importer UI, set Type to File.
   d) Drag the 201402_trip_data.csv file to the Path field as shown in the following image:



   e) Set the formats as follows:
      - Field Separator = Comma (,)
      - Record Separator = New line
      - Quote Character = Double Quote

      Then click Next at the bottom of the page.
   f) Set the properties Format = Text.
   g) Edit the FIELDS as follows:
      - Rename Bike # to Bike ID
      - Change the data type of ZipCode to string.
      - Remove all of the spaces in the Name fields.

      Then click Submit at the bottom of the page.

3. Click Query at the top of the page and select Editor Hive to open the Hive editor and then create a query.
   - Enter the following query into the editor window:

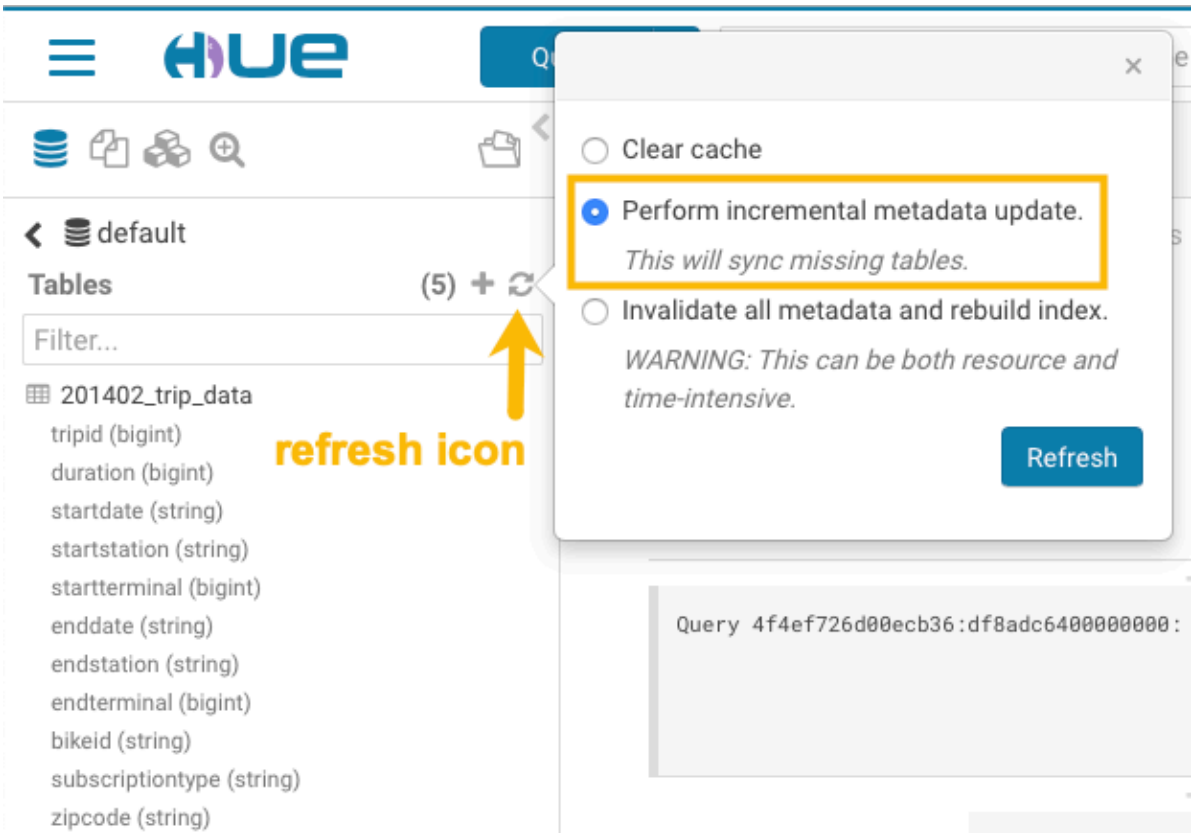
     ```
     SELECT * FROM default.201402_trip_data
     LIMIT 10;
     ```

   - Click the execute icon to run the query. The following rows are returned:

Query History | Saved Queries | Results (10)

	201402_trip_data.tripid	201402_trip_data.duration	201402_trip_data
1	4576	63	8/29/2013 14:13
2	4607	70	8/29/2013 14:42
3	4130	71	8/29/2013 10:16
4	4251	77	8/29/2013 11:29
5	4299	83	8/29/2013 12:02
6	4927	103	8/29/2013 18:54
7	4500	109	8/29/2013 13:25
8	4563	111	8/29/2013 14:02
9	4760	113	8/29/2013 17:01
10	4258	114	8/29/2013 11:33

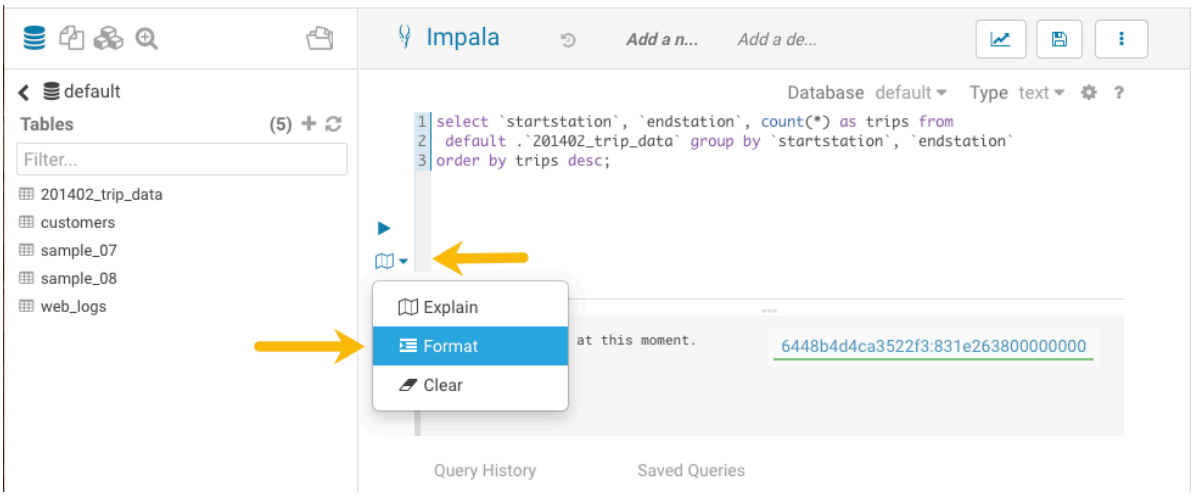
4. Click Query at the top of the page and select Editor Impala to open the Impala SQL editor and then create a query.
   a. In the left panel, click the refresh icon and select Perform incremental metadata update to make the new table visible to Impala:



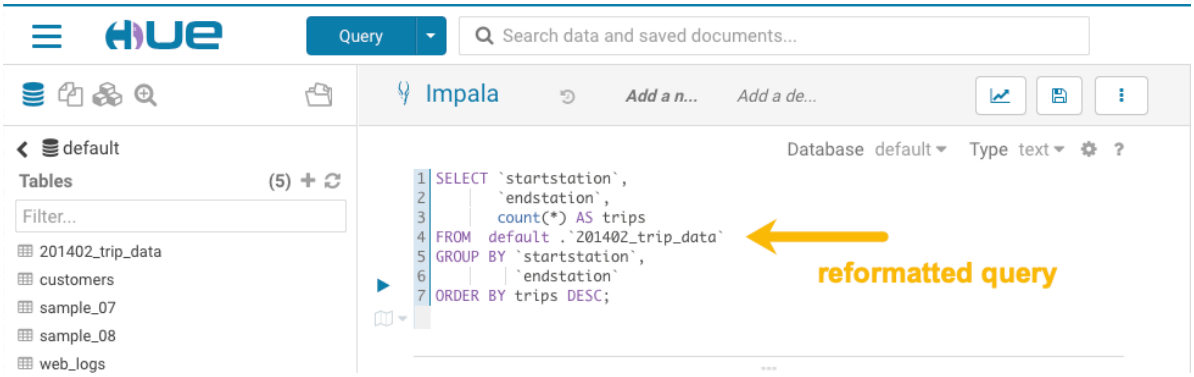
   b. Enter the following query into the editor window:

```
select 'startstation', 'endstation', count(*) as trips from default.'20
1402_trip_data'
group by 'startstation', 'endstation' order by trips desc;
```

c. Click the down arrow just under the execution icon and select Format:



This reformats the query:

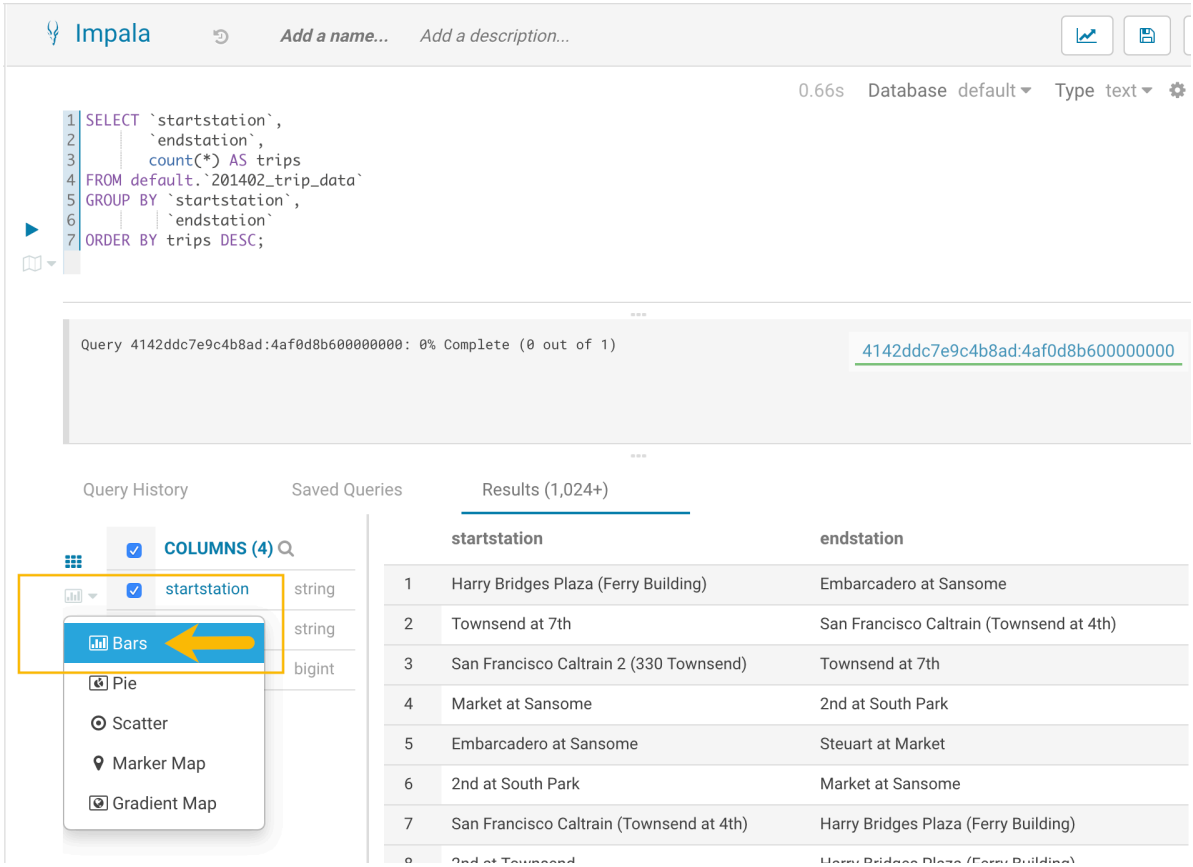


d. Click the save icon , enter a query name, and click Save.

e. Click the execute icon to run the query.

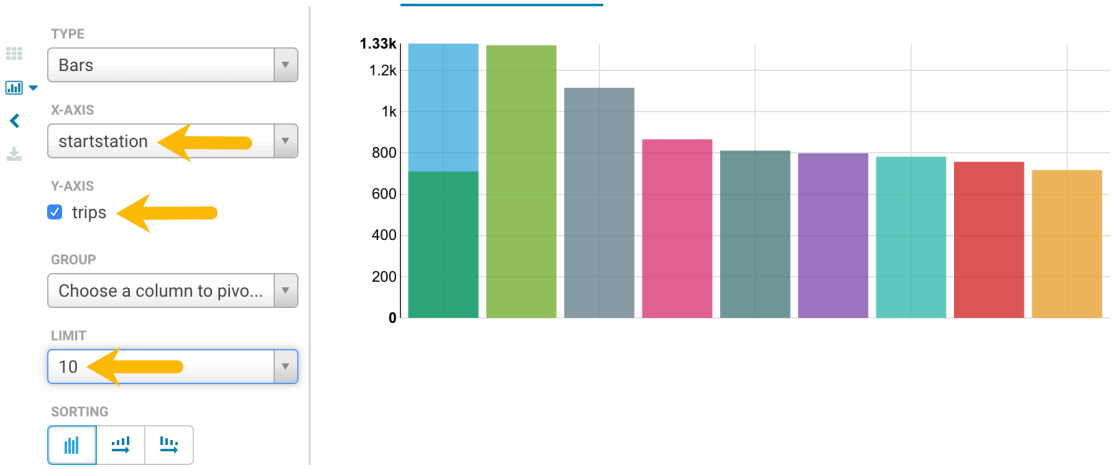
5. Create a bar chart that is based on the query results:

   a. Click the chart icon and then select Bars.



   b. Set the bar chart elements as follows:

   - X-AXIS = startstation
   - Y-AXIS = trips
   - LIMIT = 10



6. Create a pie chart by clicking the chart icon again and then select Pie.
7. Download the query results by clicking and selecting in what format you want to download, copy, or export the results.

Viewing Hive query history

The Job Browser page displays all the queries that were run on the Hive Virtual Warehouse from various query interfaces, such as Beeline, Hive Warehouse Connector (HWC), Tableau, and so on.

About this task

Note: This feature is available only for Hive queries.

Only Query Store Administrators can view historical queries of all users to monitor resource utilization and control costs from the Hue Job Browser. Non-admin users can view only their queries.

The Hue Query Store does not clean up automatically. The queries are retained in the backend database until you clean up the records. You can view queries up to five years old from the Hue web interface.

Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.

   The **Job Browser** page is displayed.
3. Click Queries.

   The Hive queries that were run by all users for the past seven days are displayed. You can select the time period for which you want to view the historical data.

   You can also filter queries by their status.

Related Information

Adding Query Store Administrator users on Data Hub clusters

Viewing Hive query information

The Query Info tab provides information such as, the Hive query ID, the user who executed the query, the start time, the end time, the total time taken to execute the query, the tables that were read and written, application ID, Directed Acyclic Graph (DAG) IDs, session ID, LLAP app ID, thread ID, and the queue against which the query was run.

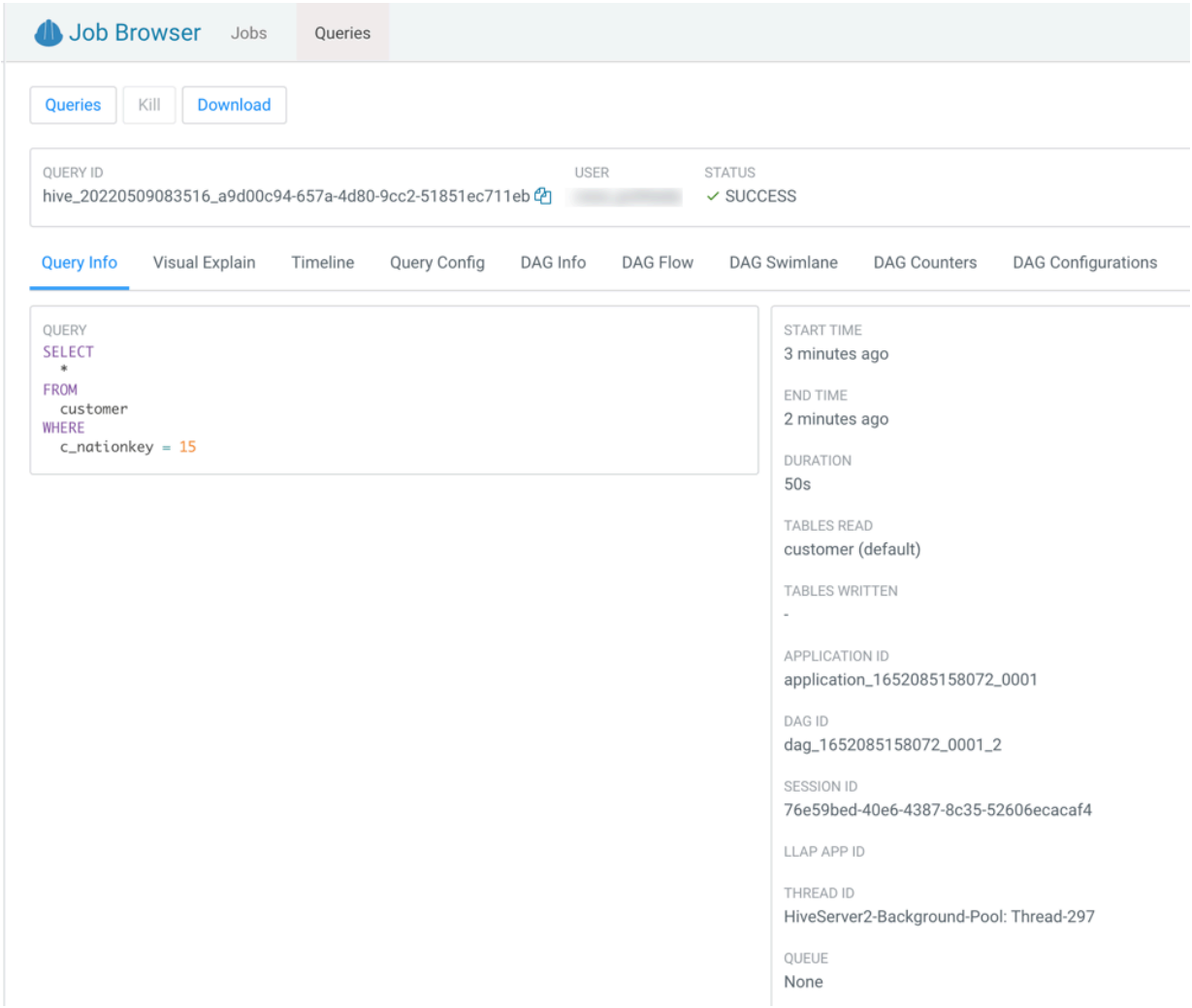
About this task

Note: This feature is available only for Hive queries.

Procedure

1. Log in to the Hue web interface.

2. Click Jobs from the left assist panel.
   The **Job Browser** page is displayed.
3. Go to the **Queries** tab and click on the query for which you want to view the query details.
   The following image shows the **Query Info** tab on the Hue web interface:



Viewing explain plan for a Hive query

The Visual Explain feature provides a graphical representation of the query execution plan. The Explain plan is read from right to left. It provides details about every stage of query execution.

About this task

Note: This feature is available only for Hive queries.

Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.
   The **Job Browser** page is displayed.

3. Go to the **Queries** tab and click on the query for which you want to view the query details.
4. Click on Visual Explain.

   The following image shows the **Visual Explain** tab on the Hue web interface:

5. (Optional) Click to download the query explain plan in JSON format.

Viewing Hive query timeline

The Timeline tab provides a visual representation of Hive performance logs and shows the time taken by each stage of the query execution.

About this task

Note: This feature is available only for Hive queries.

Following are the stages in which a query is executed:

- Pre-execution and DAG construction: It is the first phase of query execution and is executed on the Hive engine. It constitutes the time taken to compile, parse, and build the Directed Acyclic Graph (DAG) for the next phase of the query execution.
- DAG submission: It is the second phase in which the DAG that was generated in Hive is submitted to the Tez engine for execution.
- DAG runtime: It shows the time taken by the Tez engine to execute the DAG.
- Post-execution: It is the last phase of query execution in which the files in S3/ABFS are moved or renamed.

Duration data about each phase are distilled into more granular metrics based on query execution logs.

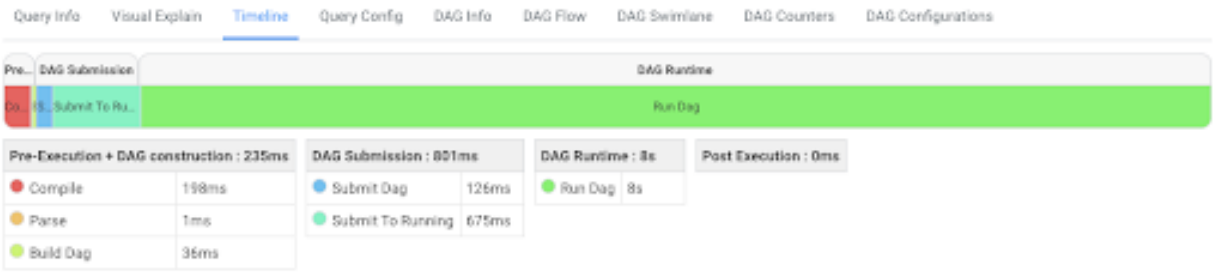
Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.

   The **Job Browser** page is displayed.
3. Go to the **Queries** tab and click on the query for which you want to view the query details.

4. Click on Timeline.

   The following image shows the **Timeline** tab on the Hue web interface:



Viewing configurations for a Hive query

The Query Config tab provides the configuration properties and settings that are used in a Hive query. You can use this tab to verify that configuration property values align with your expectations.

About this task

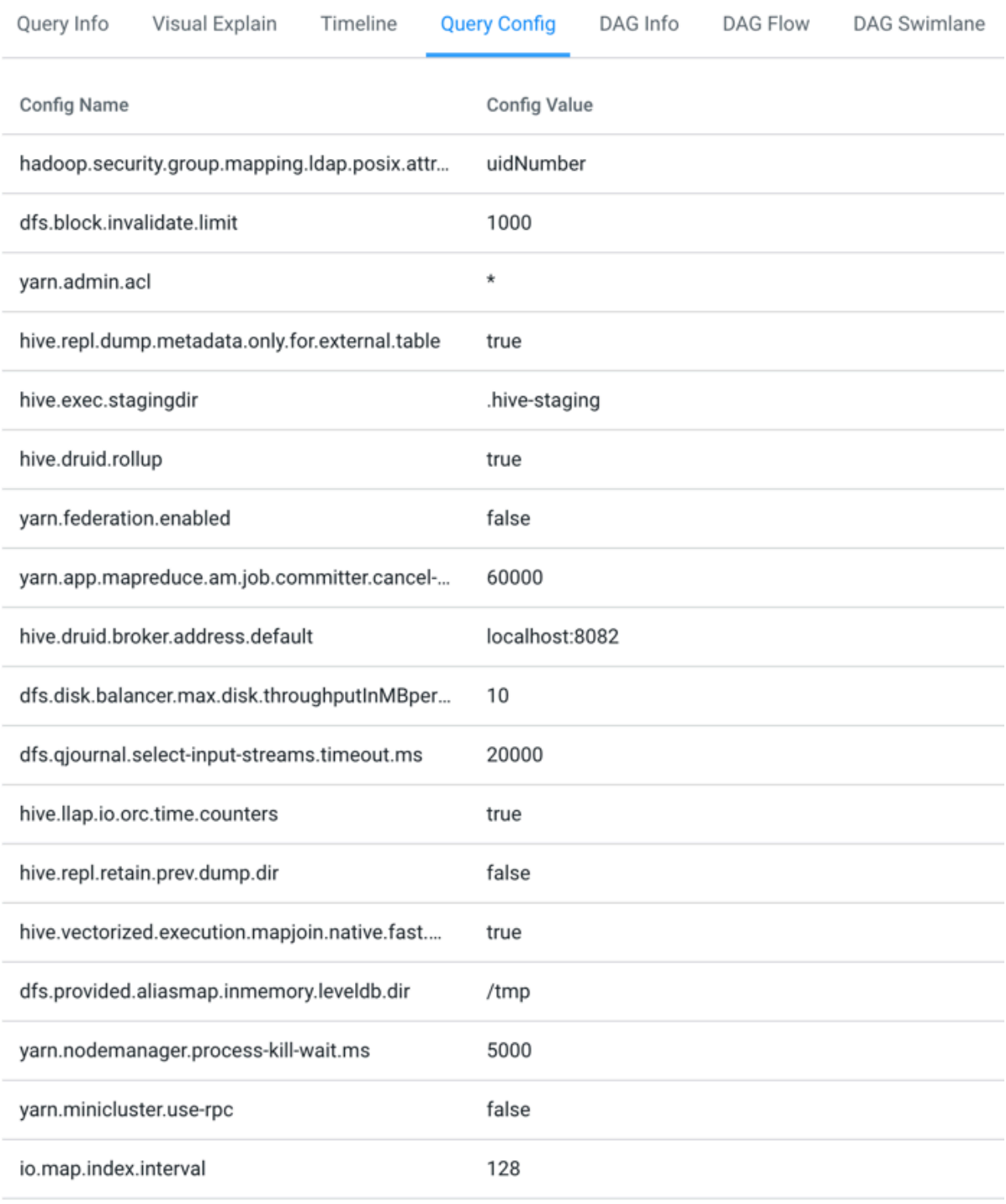
Note: This feature is available only for Hive queries.

Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.

   The **Job Browser** page is displayed.
3. Go to the **Queries** tab and click on the query for which you want to view the query details.

4. Click on Query Config.
   The following image shows the **Query Config** tab on the Hue web interface:



Viewing DAG information for a Hive query

Directed Acyclic Graph (DAG) is created by the Hive engine every time you query the Hive Virtual Warehouse. The Hive SQL queries are compiled and converted into a Tez execution graph also known as a DAG. DAG is a

collection of vertices where each vertex executes a fragment of the query or script. Hue provides a web interface to view detailed information about DAGs.

About this task

Note: This feature is available only for Hive queries.

Directed connections between vertices determine the order in which they are executed. For example, the vertex to read a table must be run before a filter can be applied to the rows of that table. As another example, consider a vertex that reads a user table that is very large and distributed across multiple computers and multiple racks. Reading the table is achieved by running many tasks in parallel.

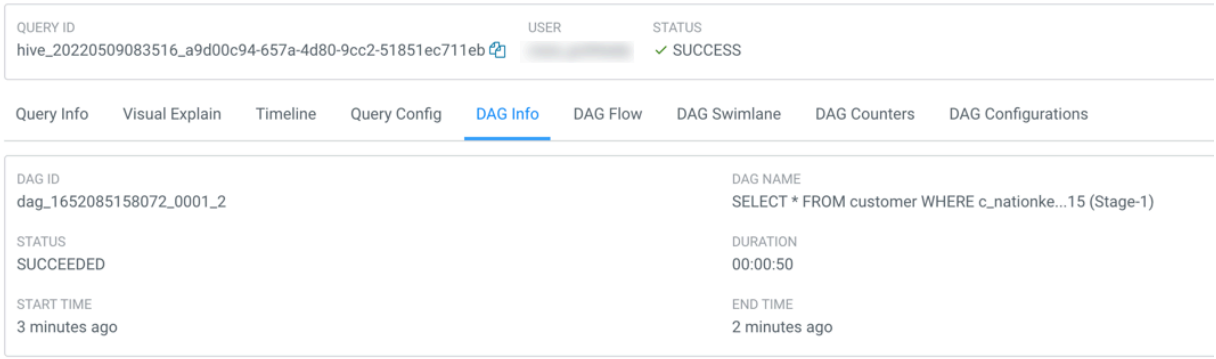
Important: The DAG information tabs (**DAG Info**, **DAG Flow**, **DAG Swimlane**, **DAG Counters**, **DAG Configurations**) are displayed only if the Tez engine is used for query execution. The Tez engine is typically utilized for complex queries.

Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.

   The **Job Browser** page is displayed.
3. Go to the **Queries** tab and click on the query for which you want to view the query details.
4. Click DAG Info to see the DAG ID, DAG name, the status of the query, the time taken to execute the DAG, start time, and end time.

   The following image shows the **DAG Info** tab on the Hue web interface:



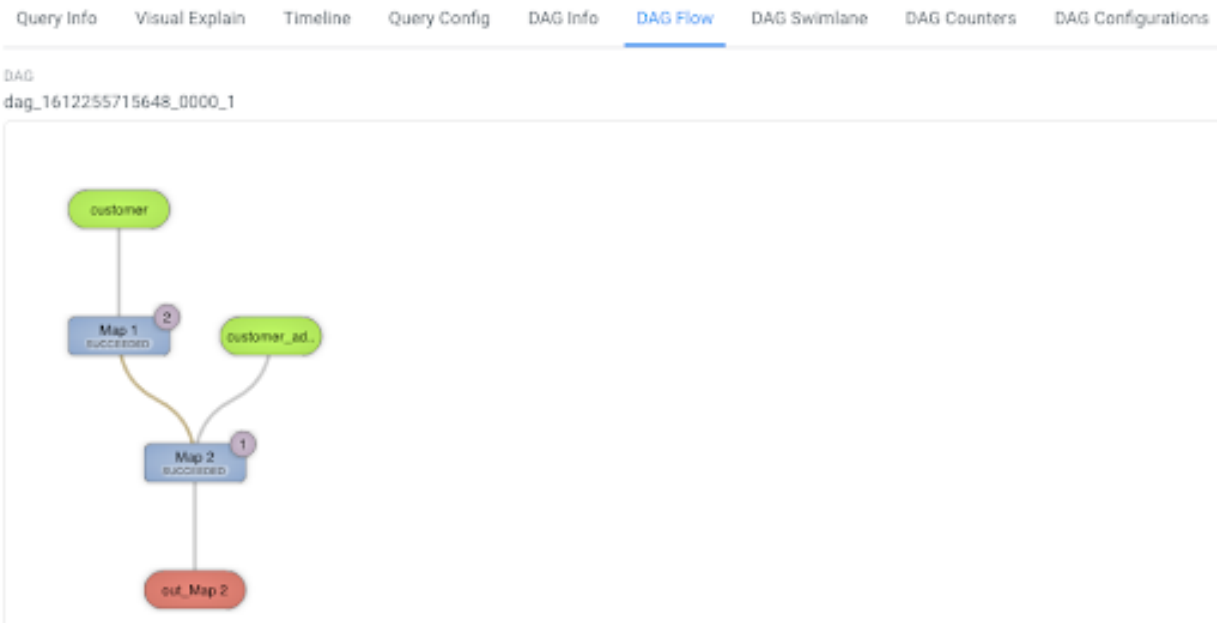
   The following table lists and describes the status of the Tez job:

Status	Description
Submitted	The DAG is submitted to Tez but is not running
Running	The DAG is currently running
Succeeded	The DAG was completed successfully
Failed	The DAG failed to complete successfully
Killed	The DAG was stopped manually
Error	An internal error occurred when executing the DAG

5. Click DAG Flow to see the DAG in the form of a flowchart.

   You can gain insight into the complexity and the progress of executing jobs, and investigate the vertices that have failures or are taking a long time to complete.

   The following image shows the **DAG Flow** tab on the Hue web interface::



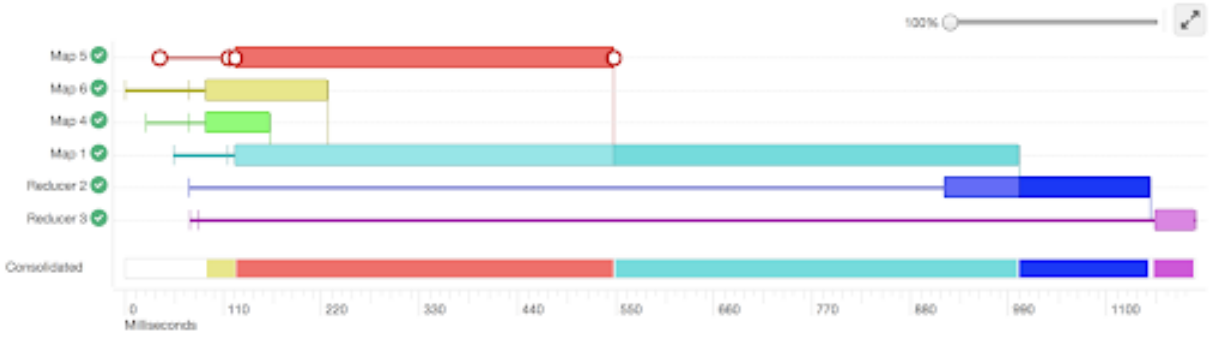
   Here, the input to vertices Map 1 and Map 2 are the tables displayed in green boxes. Next, Map 2 depends on the result set generated by Map 1. Map 2 is the last vertex in the DAG flow and after it completes its execution, the query output is written to a file in a filesystem such as S3 or ABFS.

   There are a few options to change the layout of the DAG flow. You can hide the input and the output nodes to view only the task vertices by clicking the Toggle source/sink visibility button. You can switch between the horizontal and vertical orientation by clicking the Toggle orientation button.

6. Click DAG Swimlane to see the DAG of the vertices against time.

   Each mapping and reducing task is a vertex. Each horizontal bar of the swimlane represents the total time taken by the vertex to complete the execution. The vertical lines indicate the time when the vertex was initialized, the time when the vertex started, the time when the first task started, the time when the last task was completed, and the time when the vertex finished its execution. When you mouse over the vertical line, the bubble displays the stage of the vertex execution and provides a timestamp. The vertical lines connecting two vertices denote the dependency of a vertex on another vertex.

   The following image shows the **DAG Swimlane** tab on the Hue web interface:



   In this example, Map 1 depends on the results of Map 5. Map 1 will finish its execution only when Map 5 finishes its execution successfully. Similarly, Reducer 2 depends on Map 1 to complete its execution.

   The consolidated timeline shows the percentage of time each vertex took to complete executing.

7. Click DAG Counters to see details such as the number of bytes read and written, number of tasks that initiated and ran successfully, amount of CPU and memory consumed, and so on.

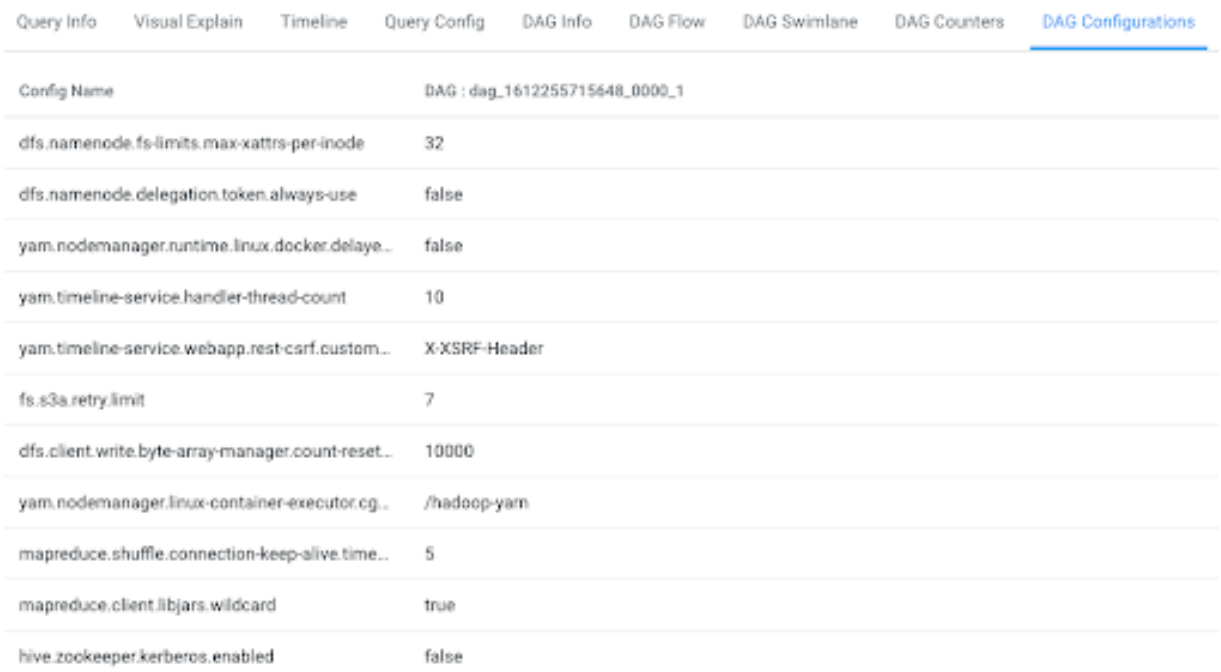
   The **DAG Counters** tab provides a way to measure the progress or the number of operations that occur within a generated DAG. Counters are used to gather statistics for quality control purposes or problem diagnosis.

   The following image shows the **DAG Counters** tab on the Hue web interface:

Query Info | Visual Explain | Timeline | Query Config | DAG Info | DAG Flow | DAG Swimlane | DAG Counters | DAG Configurations

Group Name	Counter Name	DAG : dag_1652085158072_0001_2
org.apache.tez.common.counters.DAGCounter	NUM_SUCCEEDED_TASKS	54
org.apache.tez.common.counters.DAGCounter	TOTAL_LAUNCHED_TASKS	54
org.apache.tez.common.counters.DAGCounter	DATA_LOCAL_TASKS	54
org.apache.tez.common.counters.DAGCounter	AM_CPU_MILLISECONDS	5890
org.apache.tez.common.counters.DAGCounter	AM_GC_TIME_MILLIS	22
org.apache.tez.common.counters.FileSystem...	FILE_BYTES_WRITTEN	1074416910
org.apache.tez.common.counters.FileSystem...	S3A_BYTES_READ	6742239795
org.apache.tez.common.counters.FileSystem...	S3A_READ_OPS	1124
org.apache.tez.common.counters.TaskCounter	TASK_DURATION_MILLIS	759357
org.apache.tez.common.counters.TaskCounter	INPUT_RECORDS_PROCESSED	146519
org.apache.tez.common.counters.TaskCounter	INPUT_SPLIT_LENGTH_BYTES	12387638515
HIVE	CREATED_FILES	36
HIVE	RECORDS_IN_Map_1	150000000
HIVE	RECORDS_OUT_0	6003115
HIVE	RECORDS_OUT_OPERATOR_FIL_5	6003115
HIVE	RECORDS_OUT_OPERATOR_FS_7	6003115
HIVE	RECORDS_OUT_OPERATOR_SEL_6	6003115
HIVE	RECORDS_OUT_OPERATOR_TS_0	150000000

8. Click DAG Configurations to see the Tez configuration details for a query that has a DAG associated with it.
   The following image shows the **DAG Configurations** tab on the Hue web interface:



Comparing Hive queries in Hue

You can compare two queries to know how each query is performing in terms of speed and cost-effectiveness. Hue compares various aspects of the two queries, based on which you can identify what changed between the executions of those two queries, and you can debug performance-related issues between different runs of the same query.

About this task

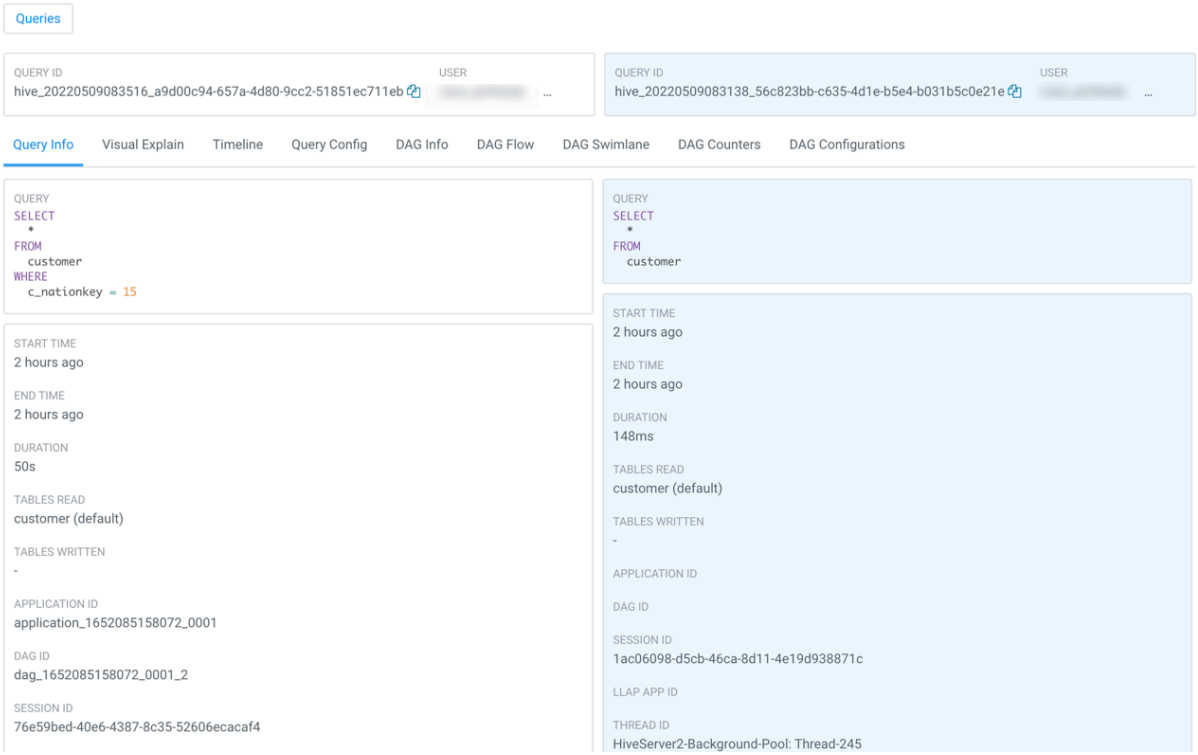
Note: This feature is available only for Hive queries.

The query comparison report provides you a detailed side-by-side comparison of your queries, including recommendations for each query, metadata about the queries, visual explain for each query, query timeline, query configuration, Directed Acyclic Graph (DAG) information, DAG flows, DAG swimlanes, DAG counters, and DAG configurations.

Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.
   The **Job Browser** page is displayed.
3. Go to the **Queries** tab.
   A list of queries that were run is displayed.

4. Select the two queries you want to compare and click Compare.
   The query comparison report is displayed, as shown in the following image:



Terminating Hive queries

If a query is running for longer than expected, or you have accidentally triggered it, then you can stop the query to free up the resources. Hue also allows you to stop multiple queries at once.

About this task

Note: This feature is available only for Hive queries. Only admin users or Hue superusers can stop running queries.

Procedure

1. Log in to the Hue web interface.
2. Click Jobs from the left assist panel.
   The **Job Browser** page is displayed.
3. Go to the **Queries** tab.
   A list of queries that were run is displayed.
4. Select the queries that you want to stop and click Kill.

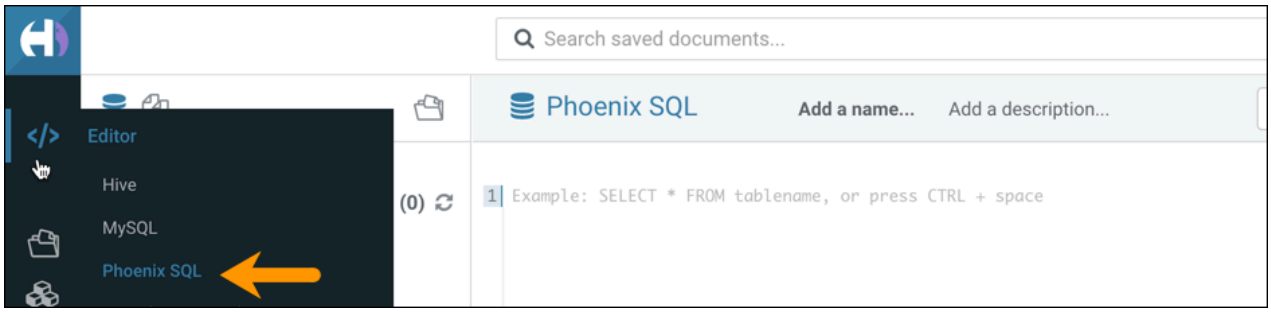
Using SQL to query HBase from Hue

Hue provides a simple SQL interface to create and manipulate SQL tables that are stored in HBase, and define and manipulate views on HBase tables using Apache Phoenix in addition to HBase shell and database API.

Cloudera does not recommend manipulating Phoenix tables from HBase as this can lead to data loss.

The SQL connector is shipped with Hue so that you do not have to download and configure it yourself.

Following are some examples to create and manipulate the Phoenix SQL tables from the Hue editor:



Creating a table

```
CREATE TABLE IF NOT EXISTS Company (company_id INTEGER PRIMARY KEY, name VAR
CHAR(225));
```

Upserting values in the table

```
UPSERT INTO Company VALUES(1, 'Cloudera');
UPSERT INTO Company VALUES(2, 'Apache');
```

Querying the table

```
SELECT * FROM Company;
```

Deleting a record

```
DELETE FROM Company WHERE COMPANY_ID=1;
```

Dropping the table

```
DROP TABLE Company;
```

Related Information

Apache Phoenix SQL command reference

Apache Phoenix Grammar

Querying existing HBase tables

To use SQL for querying data from existing HBase tables, you must create a view in Phoenix pointing to the HBase table.

To map the existing tables to the views, run the following statement using the Phoenix editor on the Hue web interface:

```
CREATE VIEW if not exists "[***HBASE-TABLE-NAME***]" ( pk VARCHAR PRIMARY KE
Y, val VARCHAR );
```

Related Information

Mapping a Phoenix table to an existing HBase table

Enabling the SQL editor autocompleter

Autocompleter provides finely tuned SQL suggestions for Hive and Impala dialects while you enter queries into the editor window. See Brand new Autocompleter for Hive and Impala in the Hue blog.

About this task

Autocompleter is enabled by default. To manually enable or disable it, open the editor configuration panel and edit settings as follows:

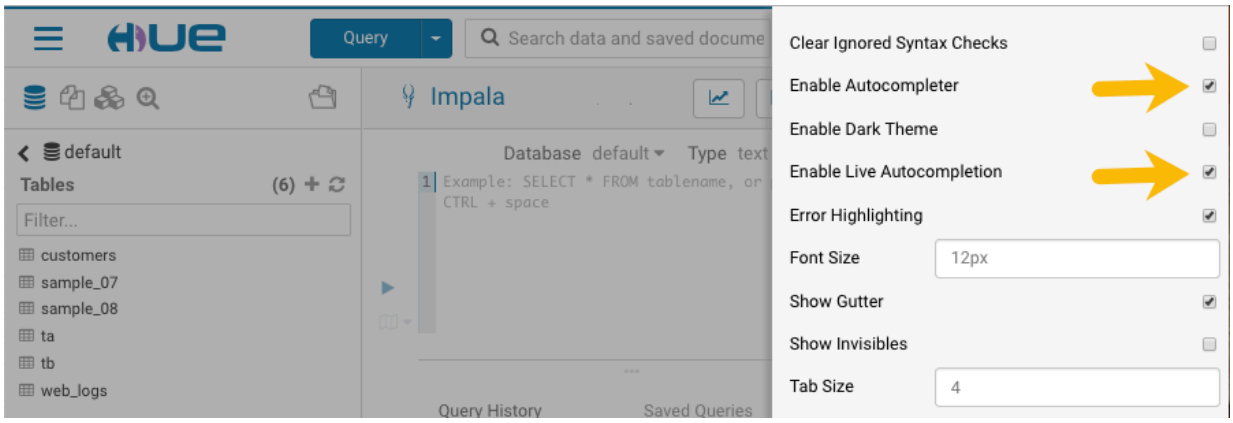
Procedure

1. Log in to Hue and go to either the Hive or Impala editor.
2. Place your cursor in the editor window and then use one of the following keyboard shortcuts to open the editor configuration panel:
   - On a Mac system, use the Command key followed by a hyphen and then a comma:

     Command-,
   - On a Windows system, use the Ctrl key followed by a hyphen and then a comma:

     Ctrl-,

   Tip: Type a question mark (?) anywhere but in the active editor window to open a menu of editor keyboard shortcuts.
3. To enable autocompletion, check the box adjacent to Enable Autocompleter. When you check Enable Autocompleter, Enable Live Autocompletion is automatically enabled as well. Place your cursor in the editor window to close the configuration panel.



4. To disable autocompletion:
   - Uncheck Enable Live Autocompletion but leave Enable Autocompleter checked, and then place your cursor in the editor window to close the configuration panel. This disables live autocompletion, but if you want to use autocompletion while building your queries in the editor, enter the following key stroke sequence to activate autocompletion: Ctrl + Space Key
   - Uncheck both Enable Autocompleter and Enable Live Autocompletion, and then click in the editor to close the configuration panel. This disables all autocompletion functionality.

Rerunning a query from the Job Browser page

You may need to reconfigure the query parameters post a query execution, based on the results. You can modify and rerun your historical queries from the Job Browser page.

About this task

Note: Only Query Processor Administrators can view and rerun historical queries of all users from the **Job Browser** page. Non-admin users can view and rerun only their queries.

Procedure

1. Log in to the Hue web interface.
2. Select Jobs from the left assist panel and click Queries on the **Job Browser** page.
3. Click on the query that you want to rerun.
   The query details are displayed.
4. Click Re Execute.
   You are redirected to the query editor where you can make the required changes to the pre-populated query.
5. Click to submit the query.

Using governance-based data discovery

Hue can use the metadata tagging, indexing, and search features available in Apache Atlas data management. After integrating Hue with Atlas, classifications and indexed entities can be accessed and viewed in Hue. This topic shows you how to use metadata classifications in Hue.

Integration between Hue and Atlas is enabled by default, but if your administrator has disabled it, it must be re-enabled before you can use governance-based data discovery.

You can create tags to classify your data both from Atlas and Hue.

Searching metadata tags

The SQL Editor in Hue provides a search text box where you can search on the metadata tags or classifications that are associated with your databases, tables, and columns.

About this task

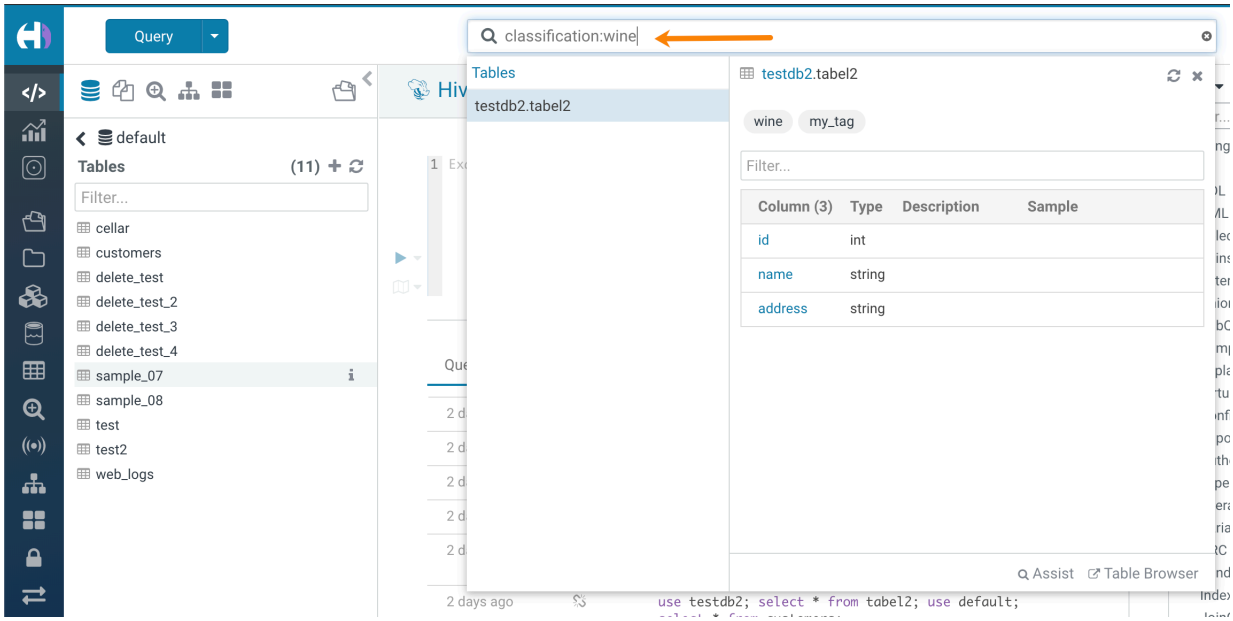
You can search for tags or classifications in either the Hive or the Impala editors.

Note: On clusters that use Apache Ranger for role-based access control, the Search mechanism does not display counts of popular values. Ranger ensures that Hue users can view only entities to which their user role (as configured and managed by Ranger) has been granted specific permissions.

Procedure

1. Go to Query Editor Impala or Hive.

2. To locate the tags or classifications in Apache Atlas, in the metadata search box located just to the right of the Query drop-down menu, type a tag: or classification: facet followed by its name. For example, type classification: wine as shown in the following image:



After you type the search facet and the tag or classification name in the search box, the <*DATABASE*>.<*TABLE*> where the tag or classification is found is returned. Click the <*DATABASE*>.<*TABLE*> to view the tags and classifications that have been defined for it.

Using Amazon S3 with Hue

Hue can read to and write to an Amazon S3 bucket.

Note:

You can use the S3 browser to store and retrieve data from Amazon S3 buckets into Hue and vice versa. If your CDP cluster is secured using Knox, then the Hue users will be able to access the S3 folders as per the access rules set on the individual folders. The access permissions are governed by IDBroker.

Only Hue superusers can view and access the S3 browser.

In CDP, you can enable the S3 File Browser in Hue in the following three ways:

- With IDBroker
- Without IDBroker
- With Ranger Authorization Service

Enabling S3 browser for Hue configured with IDBroker

You can access the S3 buckets from Hue to upload files and tables to S3 and import CSV files as tables directly in Hue by enabling the S3 browser on the Hue user interface.

Before you begin

If you have set up authentication using Knox IDBroker on your cluster, then Hue automatically detects and uses the IDBroker mappings from your cluster's core-site.xml file. Verify that the following property is present in the core-site.xml file:

```
<property>
  <name>fs.s3a.ext.cab.address</name>
  <value>https://<IDBROKERURL>:8444/gateway</value>
</property>
```

This property is automatically appended to the core-site.xml file when you enable Knox IDBroker on your cluster.

Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue service Configuration .
3. Enter the following in the Hue Server Advanced Configuration Snippet (Safety Valve) for hue_safety_valve_server.ini field:

   ```
   [desktop]
   # Remove the file browser from the blacklisted apps.
   # Tweak the app_blacklist property to suit your app configuration.
   app_blacklist=spark,zookeeper,hive,hbase,search,oozie,jobsub,pig,sqoop,sec
   urity
   [aws]
   has_iam_detection=true
   [[aws_accounts]]
   [[[default]]]
   region=[***AWS-REGION***]
   # Set a particular S3 bucket as the default
   [filebrowser]
   remote_storage_home=s3a://[***S3-BUCKET-NAME***]
   ```

   The custom configuration is stored in the hue_safety_valve_server.ini file.
4. Click Save Changes.
5. Restart the Hue service.

   The S3 file browser icon appears on the left Assist panel as well as on the left navigation bar on the Hue web interface.

What to do next

You must manually grant the following application permission to non-admin users and groups for them to be able to view and access S3 File Browser in Hue: filebrowser.s3_access:Access to S3 from filebrowser and filepicker..

You must also add the CDP users and groups to IAM role mappings.

Related Information

Granting permission to access S3 and ABFS File Browser in Hue

Adding CDP user/group to IAM role mappings

Enabling S3 browser for Hue configured without IDBroker

You can access the S3 buckets from Hue to upload files and tables to S3 and import CSV files as tables directly in Hue by enabling the S3 browser. You must specify the AWS access key and secret access key information along with the AWS region in the Hue Server Advanced Configuration Snippet if you have not set up authentication using Knox IDBroker.

Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Hue service Configuration .
3. Enter the following in the Hue Server Advanced Configuration Snippet (Safety Valve) for hue_safety_valve_server.ini field:

```
[desktop]
# Remove the file browser from the blacklisted apps.
# Tweak the app_blacklist property to suit your app configuration.
app_blacklist=spark,zookeeper,hive,hbase,search,oozie,jobsub,pig,sqoop,sec
urity
[aws]
[[aws_accounts]]
[[[default]]]
access_key_id=[***AWS-ACCESS-KEY***]
secret_access_key=[***SECRET-ACCESS-KEY***]
region=[***AWS-REGION***]

# Set a particular S3 bucket as the default
[filebrowser]
remote_storage_home=s3a://[***S3-BUCKET-NAME***]
```

   The custom configuration is stored in the hue_safety_valve_server.ini file.
4. Click Save Changes.
5. Restart the Hue service.

   The S3 file browser icon appears on the left Assist panel as well as on the left navigation bar on the Hue web interface.

What to do next

You must manually grant the following application permission to non-admin users and groups for them to be able to view and access S3 File Browser in Hue: filebrowser.s3_access:Access to S3 from filebrowser and filepicker.

Related Information

Granting permission to access S3 and ABFS File Browser in Hue

Populating an S3 bucket

Use the Hue Web UI to populate buckets in Amazon S3.

About this task

Use open data from the U.S. Geological Survey to demonstrate how to populate and S3 bucket with Hue.

Procedure

1. Download 30 days of earthquake data (all_month.csv) from the USGS (~2 MB).
2. Log on to the Hue Web UI from Cloudera Manager.
3. Select File BrowserS3 Browser.
4. Click NewBucket, name it "quakes_<any unique id>" and click Create.

   Tip: Unique bucket names are important per S3 bucket naming conventions.
5. Navigate into the bucket by clicking the bucket name
6. Click NewDirectory, name it "input" and click Create.
7. Navigate into the directory by clicking the directory name.

8. Click Upload and select, or drag, all_month.csv. The path is s3a://quakes/input/all_month.csv.

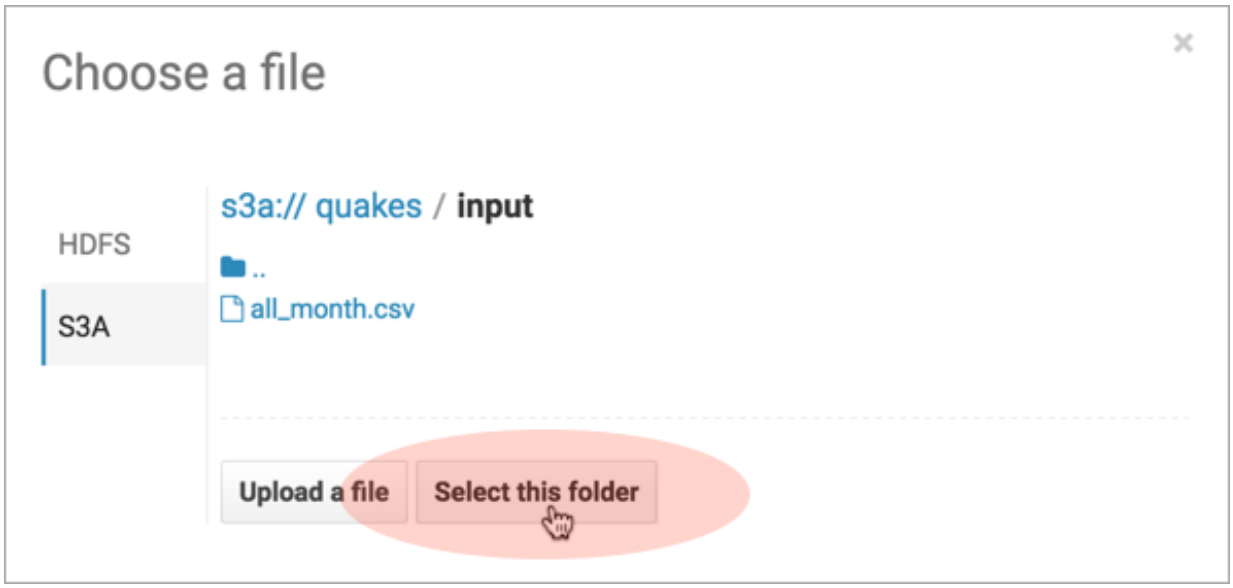
   Important: Do not add anything else to the "input" directory–no extra files, no directories.

Creating a table from an Amazon S3 file

Using Hue to create a table from an Amazon S3 file streamlines the process.

Procedure

1. Go to the Metastore Manager by clicking Data BrowsersMetastore Tables.
2. Create a new table from a file by clicking .
3. Enter a Table Name such as "earthquakes".
4. Browse for the Input Directory, s3a://quakes/input/, and click Select this folder.



5. Select Create External Table from the Load Data menu and click Next.
6. Delimit by Comma(,) and click Next.
7. Click Create Table.
8. Click the Browse Data icon to automatically generate a SELECT query in the Hive editor:

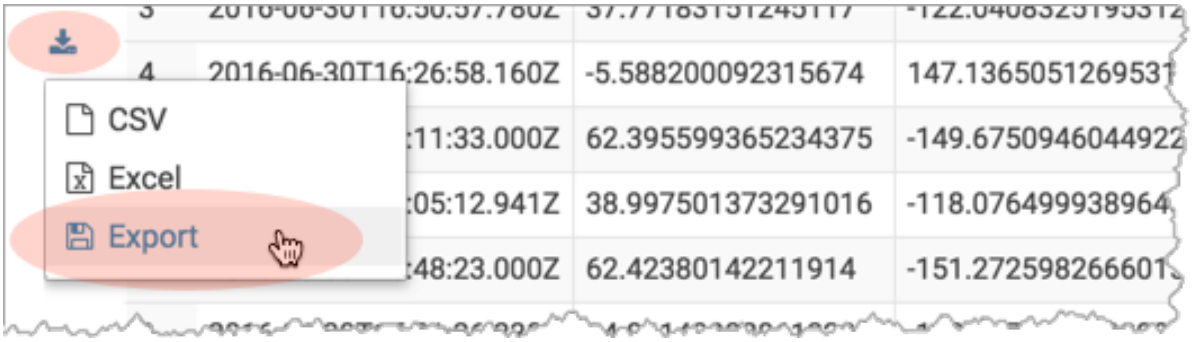
```
SELECT * FROM `default`.`earthquakes` LIMIT 10000;
```

Exporting query results to Amazon S3

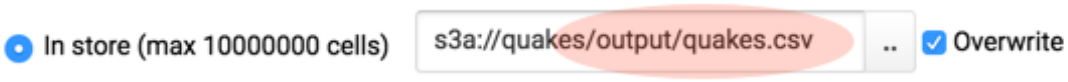
Use Hue to export query results to Amazon S3 as a custom file, a MapReduce file, or as a table.

Procedure

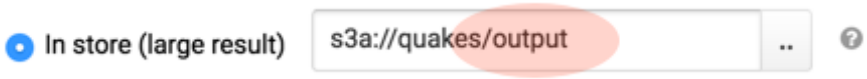
1. Run and Export Results in Hive
   a) Run the query by clicking Execute .
   b) Click Get Results .
   c) Select Export to open the Save query result dialog.



2. Save Results as Custom File
   a) Select In store (max 10000000 cells) and open the Path to CSV file dialog.
   b) Navigate into the bucket, s3a://quakes.
   c) Create folder named, "output."
   d) Navigate into the output directory and click Select this folder.
   e) Append a file name to the path, such as quakes.cvs.
   f) Click Save. The results are saved as s3a://quakes/output/quakes.csv.



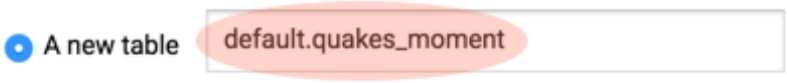
3. Save Results as MapReduce files
   a) Select In store (large result) and open the Path to empty directory dialog.
   b) Navigate into the bucket, s3a://quakes.
   c) If you have not done so, create a folder named, "output."
   d) Navigate into the output directory and click Select this folder.
   e) Click Save. A MapReduce job is run and results are stored in s3a://quakes/output/.



4. Save Results as Table
   a) Run a query for "moment" earthquakes and export:

```
SELECT time, latitude, longitude, mag
FROM `default`.`earthquakes`
WHERE magtype IN ('mw','mwb','mwc','mwr','mww');
```

   b) Select A new table and input <database>.<new table name>.
   c) Click Save.
   d) Click Browse Data to view the new table.



Using Azure Data Lake Storage Gen2 with Hue

Hue can read to and write to an Azure Data Lake Storage (ADLS) Gen2.

Note:

You can use the ABFS file browser to store and retrieve data from ADLS Gen2 into Hue and vice versa. If your CDP cluster is secured using Knox, then the Hue users will be able to access the ADLS as per the access rules set on the individual components. The access permissions are governed by IDBroker.

Only Hue superusers can view and access the ABFS file browser.

In CDP, you can enable the ABFS File Browser in Hue in the following three ways:

- With IDBroker
- Without IDBroker
- With Ranger Authorization Service

Enabling ABFS file browser for Hue configured with IDBroker

You can directly access the Azure file storage from Hue to upload files and tables to Azure file storage and import CSV files as tables directly in Hue by enabling the ABFS file browser on the Hue user interface.

Before you begin

If you have set up authentication using Knox IDBroker on your cluster, then Hue automatically detects and uses the IDBroker mappings from your cluster's core-site.xml file. Verify that the following property is present in the core-site.xml file:

```
<property>
  <name>fs.azure.ext.cab.address</name>
  <value>https://<IDBROKERURL>:8444/gateway</value>
</property>
```

This property is automatically appended to the core-site.xml file when you enable Knox IDBroker on your cluster.

You can also check whether there is a mapped role for the group(s) associated with the authenticated user by going to Management Console Environments Actions Manage Access IDBroker Mappings . If no role is mapped, then you may see the following error: Failed to obtain storage credentials from IDBroker with error: 403 Client Error: Forbidden for url: <url> { "error": "There is no mapped role for the group(s) associated with the authenticated user.", "auth_id": "<user-id>" }. To resolve this, add the user and the Data Access role to the current mappings on the **IDBroker Mappings** page.

Procedure

1. Sign in to Cloudera Manager as an Admin user.
2. Go to Clusters Hue service Configuration .
3. Specify the following in the Hue Server Advanced Configuration Snippet (Safety Valve) for hue_safety_valve_server.ini field:

   ```
   [desktop]
   # Remove the file browser from the blacklisted apps.
   # Tweak the app_blacklist property to suit your app configuration.
   app_blacklist=spark,zookeeper,hive,hbase,search,oozie,jobsub,pig,sqoop,sec
   urity
   [azure]
   [[abfs_clusters]]
   [[[default]]]
     fs_defaultfs=abfs://<CONTAINER_NAME>@<STORAGE_NAME>.dfs.core.windows.net
    webhdfs_url=https://<STORAGE_NAME>.dfs.core.windows.net/
   ```

   The custom configuration is stored in the hue_safety_valve_server.ini file.
4. (Optional) If the Knox server and Hue are not present on the same host, then add the add the Hue server's FQDN to the trusted origins property in the Hue Server Advanced Configuration Snippet (Safety Valve) for hue_safety_valve_server.ini field:

   ```
   [desktop]
   [[session]]
   trusted_origins=[***HUE-SERVER-FQDN***]
   ```
5. Click Save Changes.
6. Restart the Hue service.

   The ABFS file browser icon appears on the left Assist panel as well as on the left navigation bar on the Hue web interface.

What to do next

You must manually grant the following application permission to non-admin users and groups for them to be able to view and access ABFS File Browser in Hue: filebrowser.abfs_access:Access to ABFS from filebrowser and file picker.

You must also add the CDP users and groups to IAM role mappings.

Related Information

Granting permission to access S3 and ABFS File Browser in Hue
Adding CDP user/group to IAM role mappings

Enabling ABFS file browser for Hue configured without IDBroker

You can directly access the Azure file storage from Hue to upload files and tables to Azure file storage and import CSV files as tables directly in Hue by enabling the ABFS file browser on the Hue user interface. You must specify the Azure client ID and the client secret ID information along with the tenant ID in the Hue Server Advanced Configuration Snippet if you have not set up authentication using Knox IDBroker.

Procedure

1. Sign in to Cloudera Manager as an Admin user.
2. Go to Clusters Hue service Configuration .
3. Specify the following in the Hue Server Advanced Configuration Snippet (Safety Valve) for hue_safety_valve_server.ini field:

```
[desktop]
# Remove the file browser from the blacklisted apps.
# Tweak the app_blacklist property to suit your app configuration.
app_blacklist=spark,zookeeper,hive,hbase,search,oozie,jobsub,pig,sqoop,sec
urity
[azure]
  [[azure_accounts]]
    [[[default]]]
      client_id=<CLIENT_ID>
      client_secret=<CLIENT_SECRET_ID>
      tenant_id=<TENANT_ID>

    [[abfs_clusters]]
      [[[default]]]
  fs_defaultfs=abfs://<CONTAINER_NAME>@<STORAGE_NAME>.dfs.core.windows.net
  webhdfs_url=https://<STORAGE_NAME>.dfs.core.windows.net/
```

   The custom configuration is stored in the hue_safety_valve_server.ini file.
4. Click Save Changes.
5. Restart the Hue service.

   The ABFS file browser icon appears on the left Assist panel as well as on the left navigation bar on the Hue web interface.

What to do next

You must manually grant the following application permission to non-admin users and groups for them to be able to view and access S3 File Browser in Hue: filebrowser.abfs_access:Access to ABFS from filebrowser and filepicker.

Related Information

Granting permission to access S3 and ABFS File Browser in Hue

Granting permission to access S3 and ABFS File Browser in Hue

Only admin users can view and access S3 or ABFS File Browser in Hue after enabling it. You must manually grant application permissions to non-admin users and groups for them to be able to view and access S3 and ABFS File Browsers in Hue.

About this task

The following table lists the application permissions for each cloud storage type:

Cloud storage	Hue application permission
S3	filebrowser.s3_access:Access to S3 from filebrowser and filepicker.
ABFS (ADLS Gen2)	filebrowser.abfs_access:Access to ABFS from filebrowser and filepicker.
ADLS (Gen1)	filebrowser.adls_access:Access to ADLS from filebrowser and filepicker.
GS	Access to GS from filebrowser and filepicker.

Before you begin

You can only assign Hue application permissions to groups or users within a group. Add the users to a group to whom you want to grant access to the S3 or ABFS File Browsers.

Important: The "default" group in Hue does not have the permissions required to view S3 or ABFS File Browsers, by default. If your users belong to the default group, then you must manually grant the required permissions.

Procedure

1. Sign in to Hue as an Administrator.
2. Go to admin Manage Users Groups .
3. Click on the group to whom you want to grant the filebrowser application permissions.
4. On the **Edit [***GROUP-NAME***]** page, select the required permission under the permission section and click Update group.

Supported non-ASCII and special characters in Hue

Auto-generated files may often introduce non-alphanumeric characters in the file and directory names that Hue does not support. This might cause the files or directories to not appear on the Hue File Browser. Review the list of non-alphanumeric, non-ASCII, diacritics (accents) characters supported in Hue for the following operations: upload, create, list in folder, view, and rename.

Table 1: Supported characters on HDFS

Special character symbol	Description	Filename support	Folder name support
~	Tilde	Fully supported	Fully supported
@	Ampersat	Fully supported	Fully supported
#	Hash	Partially supported. Not supported for rename operations.	Unsupported
$	Dollar sign	Fully supported	Fully supported
&	Ampersand	Fully supported	Fully supported
(	Left paranthesis	Fully supported	Fully supported
)	Right paranthesis	Fully supported	Fully supported
*	Asterisk	Fully supported	Fully supported
!	Exclamation mark	Fully supported	Fully supported
+	Plus	Fully supported	Fully supported

Special character symbol	Description	Filename support	Folder name support
=	Equal	Fully supported	Fully supported
:	Colon	Unsupported	Unsupported
;	Semicolon	Fully supported	Fully supported
,	Comma	Fully supported	Fully supported
.	Period	Fully supported	Fully supported
?	Question mark Not supported with Knox.	Fully supported	Fully supported
/	Forward slash	Unsupported	Unsupported
\	Backslash	Unsupported	Unsupported
'	Apostrophe or single quote	Fully supported	Fully supported
T######-ääö	Non-standard alphabets with diacritics and accents.	Fully supported	Fully supported

Table 2: Supported characters on S3

Special character symbol	Description	Filename support	Folder name support
~	Tilde	Fully supported	Fully supported
@	Ampersat	Fully supported	Fully supported
#	Hash	Partially supported. Not supported for view and rename operations.	Unsupported
$	Dollar sign	Fully supported	Fully supported
&	Ampersand	Fully supported	Fully supported
(	Left paranthesis	Fully supported	Fully supported
)	Right paranthesis	Fully supported	Fully supported
*	Asterisk	Fully supported	Fully supported
!	Exclamation mark	Fully supported	Fully supported
+	Plus	Fully supported. Not supported for create operations on RAZ-enabled environments.	Fully supported
=	Equal	Fully supported	Fully supported
:	Colon	Unsupported	Unsupported
;	Semicolon	Fully supported	Fully supported
,	Comma	Fully supported	Fully supported
.	Period	Fully supported	Fully supported
?	Question mark Not supported with Knox.	Fully supported	Partially supported. Not supported for list, upload, and view operations.
/	Forward slash	Unsupported	Unsupported
\	Backslash	Partially supported. Not supported for upload operations.	Partially supported. Not supported for upload operations.
'	Apostrophe or single quote	Fully supported	Fully supported
T######-ääö	Non-standard alphabets with diacritics and accents.	Fully supported	Fully supported

Table 3: Supported characters on ABFS

Special character symbol	Description	Filename support	Folder name support
~	Tilde	Fully supported	Fully supported
@	Ampersat	Fully supported	Fully supported
#	Hash	Partially supported. Not supported for view and rename operations.	Unsupported
$	Dollar sign	Fully supported	Fully supported
&	Ampersand	Fully supported	Fully supported
(	Left paranthesis	Fully supported	Fully supported
)	Right paranthesis	Fully supported	Fully supported
*	Asterisk	Fully supported	Fully supported
!	Exclamation mark	Fully supported	Fully supported
+	Plus	Fully supported	Fully supported
=	Equal	Fully supported	Fully supported
:	Colon	Unsupported	Unsupported
;	Semicolon	Fully supported	Fully supported
,	Comma	Fully supported	Fully supported
.	Period	Fully supported	Fully supported
?	Question mark Not supported with Knox.	Partially supported. Not supported for view and rename operations.	Partially supported. Not supported for list, rename, and view operations.
/	Forward slash	Unsupported	Unsupported
\	Backslash	Unsupported	Unsupported
'	Apostrophe or single quote	Fully supported	Fully supported
T######-ääö	Non-standard alphabets with diacritics and accents.	Fully supported	Fully supported

Options to rerun Oozie workflows in Hue

Oozie workflows consume time and resources to run. You can optimize the rerun of a failed Oozie workflow by selecting one of the following two options: “All or skip successful” or “Only failed”. These options enable you to select individual actions within your workflow that you want to rerun.

Important: The option to rerun the an Oozie workflow is enabled in the following conditions:

- In case of an external workflow, the rerun button is enabled if the workflow was run by the same user who is logged in, or if the user is a Hue superuser.
- If the workflow is created in Hue, then the rerun button is enabled only for the user who originally ran the workflow, or for the Hue superuser.

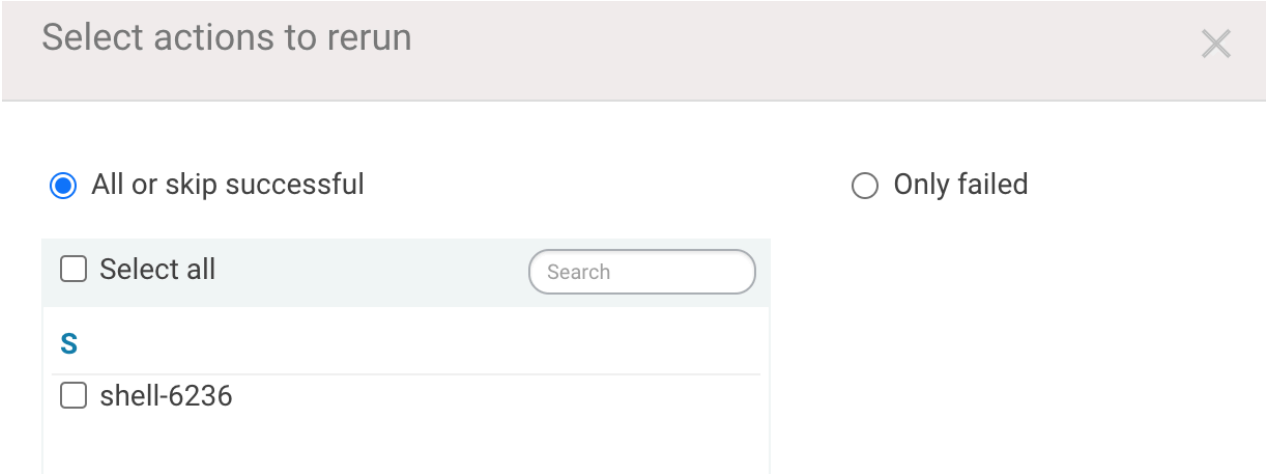
The All or skip successful option enables you to either rerun both the successful and failed actions or skip the actions that ran successfully and run only the failed ones. Using this option, you have more control over selecting the actions that you want to rerun from the list of successfully run actions.

If you select the All or skip successful option but do not select any or all of the successfully run actions, then Hue reruns the whole Oozie workflow.

If you select the All or skip successful option and select some of the successfully run actions, then Hue reruns the selected actions and the failed actions.

Select the Only failed option to only rerun the failed actions within your workflow.

The following image shows the available rerun options on the Hue web interface:



Options to restart the Hue service

Restarting the Hue service is necessary after upgrading CDP or making configuration changes to Hue. You can restart Hue in regular or rolling restart mode. Rolling restart reduces the service downtime to approximately 80 to 90 seconds.

Suppose you restart a cluster containing the Hue service in a regular mode. In that case, the Hue service can become unavailable for an average of 30 minutes from the restart time (this could vary depending on the number of services in your cluster). To reduce the service downtime, select the Rolling Restart option from the Actions menu to restart Hue. Cloudera Manager shows the following options to restart Hue in the rolling restart mode:

Category	Restart options	Description
Restart filters	Restart roles with stale configuration only	Select this option to restart Hue if you have updated the Hue configuration.
	Restart roles with the old software version only	Select this option to restart Hue if you have upgraded the cluster.
Role types to restart	Kerberos Ticket Renewer	Selected by default. Restarts the Kerberos Ticket Renewer role.
	Load Balancer	Selected by default. Restarts the Load Balancer role after the Kerberos Ticket Renewer.
	Hue Server	Selected by default. Restarts the Hue Server role after the load balancer.

When you restart Hue, Hue's non-worker roles, such as the load balancer, Kerberos ticket renewer, and Hue server restart one after the other.

Rolling restart operation is supported at the cluster level and service level.