Cloudera Runtime 7.1.2

Using Data Analytics Studio

Date published: 2020-02-28
Date modified: 2020-07-10

https://docs.cloudera.com/

Legal Notice

Contents

Compose queries

You can write and edit queries using the query composer.

Procedure

1. Select a database from the left panel.

   The Tables column displays all the tables in the database.
2. Search for the required tables, if needed.

   Click on a table name to view the columns within the table. Use the Filter field to enter text to further refine your search for the required column.
3. Enter the query in the worksheet tab.

   If your database contains more than or equal to 10,000 columns, press Ctrl + Spacebar on your keyboard to enable the auto-complete pop-up while you type the query.

   The auto-complete pop-up appears as you type if the number of columns in your database is less than 10,000.

   Each worksheet is identified by a unique name and you can add worksheets using the plus icon. As you start entering the query, the intuitive composer suggests SQL commands, keywords, and table columns according to the query.
4. Perform the desired operation on the query:
   - Click Execute to run the query. Alternatively, you can use the keyboard shortcut to execute the query: press Ctrl + Enter on Windows and control + return on XOS. Make sure that you press the keyboard shortcut while you are in the query editor.
   - Click Save As to save the worksheet with a different name.
   - Click Visual Explain to view the query details in the form of a block diagram.
   - Select Show Results if you want to view the results of the query.
   - Select Download Results to download the query results.
   - Click Saved tab to view saved queries and edit them.

Manage queries

Search for existing queries, and refine your search results based on various filters. You can also save the search for future use.

DAS uses a simple SQL query language that enables users familiar with SQL to query the data.

Searching queries

You can search for queries and see the list of queries that have been searched. You can refine your search results on the basis of parameters such as status of the query, queue to which the query belongs, the user of the query, tables read for the query, and tables written for the query, and execution modes.

Procedure

1. Enter your search query.
2. Click the time range drop down to select the from and to dates. You can also click to select one of the quick ranges provided in the list.
3. In the Refine tab, click the plus icon to further refine your search result. All options to refine your search appear. Select the required parameters and click Apply.

4. In the Actions column, click the pencil icon to open and edit the query in Composer view.
5. In the Actions column, click the i icon to view the query details.

Refining query search using filters

You can further refine your query search results using various filters.

Procedure

1. Click + in the Refine tab.
2. Select the filters from the list by clicking each filter.
3. Click Apply.

Note: The total number of items in every facet, such as User, Queue, Tables Read, etc. is displayed in parentheses next to the facet title in the Refine section. For optimum resource utilization, DAS does not refresh the result set every time you filter items. For example, if there are three users: hive, admin, and admin1, and you select only admin and click Apply, DAS does not show User (1). It still displays User (3).

Saving the search results

You have an option to save the filters with which you refined the queries, along with the result set for future use. It is easy to load, optionally modify, and run queries from the saved query list.

Click the save icon to save your search and the results. The saved query search names get listed under the SEARCHES dropdown list.

Compare queries

You can compare two queries to know how each query is performing in terms of speed and cost effectiveness. DAS 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

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 configuration, time taken at each stage of the query execution, and Directed Acyclic Graphs (DAG) information and DAG counters.

To compare two queries:

Procedure

1. Sign in to DAS and click Queries.
2. Under ACTIONS, click the Compare button to select a query.

   The selected query is displayed on the comparison toolbar.
3. Next, select the query that you want to compare by clicking the Compare button.

   The selected query is displayed on the comparison tool bar next to the query that you initially selected.
4. Click COMPARE.

   The comparison report is displayed.
5. To remove any query from the comparison toolbar, click x.
6. To change the order of the queries that are being compared, click the Swap queries button on the comparison toolbar.

View query details

For a given query, you can view the optimization recommendations, query details, the visual representation of the explain plan, Hive configurations, the time taken to compile, build and run the query, and the DAG information.

To view the query details, you can either click on a particular query or the i icon under the Actions column on the **Queries** page.

Viewing the query recommendations

The **Recommendations** tab provides information about how you can improve the performance of the query and optimize its cost.

Viewing the query details

The **Query Details** tab provides information such as, the Hive query ID of the query, 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, the application ID, the DAG ID(s), the session ID, the thread ID, and the queue against which the query was run.

Click the Edit button in the **Qury Details** section to edit and rerun the query.

Viewing the visual explain for a query

The **Visual Explain** tab provides a graphical representation of the query execution plan. The explain plan can be read from right to left. It provides details about every stage of the query operation.

Viewing the Hive configurations for a query

The **Configs** tab provides the Hive configuration details for the query. You can search for a particular configuration by entering the configuration name in the search field.

Viewing the query timeline

The **Timeline** tab shows the time taken by every stage of the query execution.

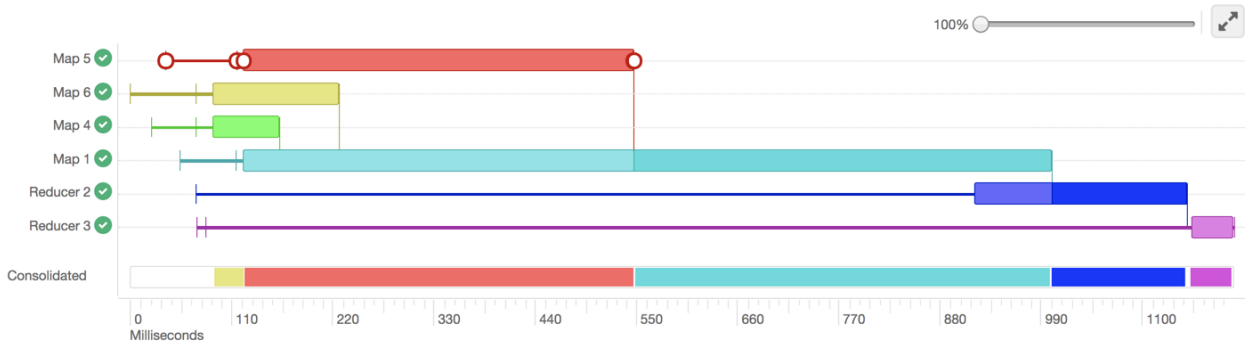
The first stage is Pre-execution and DAG construction. It is executed on the Hive engine. It constitutes the time taken to compile, parse, and build the DAG for the next phase of the query execution. In the next stage of query execution, the DAG generated in Hive is submitted to Tez engine for execution. The DAG Runtime shows the time taken by the Tez engine to execute the DAG. In the post-execution stage, the HDFS files are moved or renamed.

Viewing the task-level DAG information

The DAG Info tab shows the Directed Acyclic Graph of the vertices against time. Each mapping and reducing task is a vertex.

The following image shows the DAG swimlane:

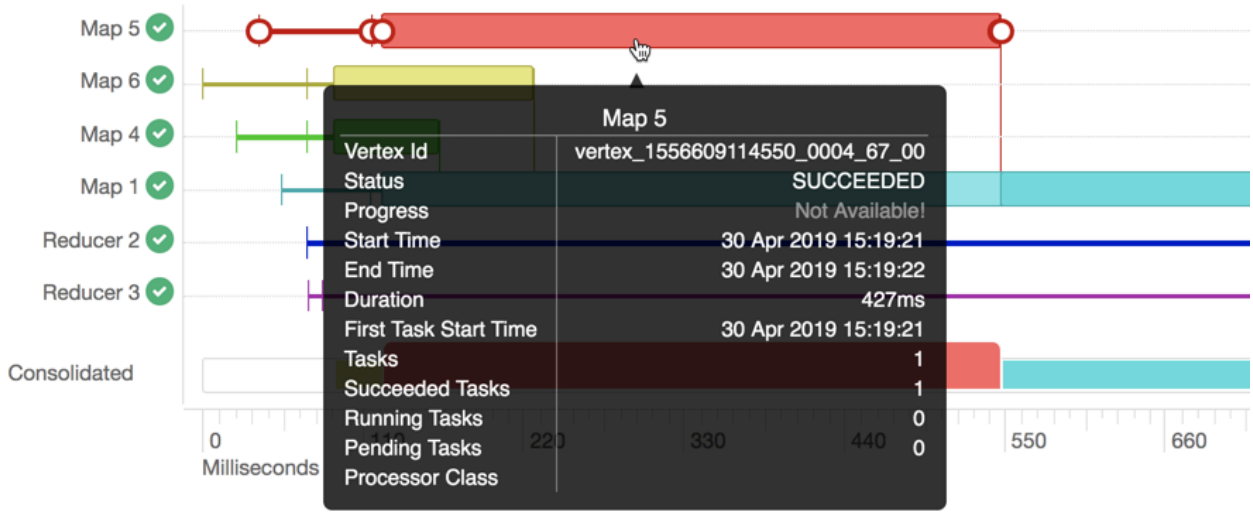
Figure 1: DAG swimlane in DAS



Each horizontal bar of the swimlane represents the total time taken by the vertex to complete. The vertical lines indicate the time when the vertex initialized, the time when the vertex started, the time when the first task started, the time when the last task completed, and the time when the vertex finished its execution. When you mouseover the vertical line, the bubble displays the stage of the vertex execution and provides a timestamp.

To know more about a particular vertex, hover the mouse pointer anywhere on the horizontal bar, as shown in the following image:

Figure 2: DAG swimlane: Viewing information about a vertex

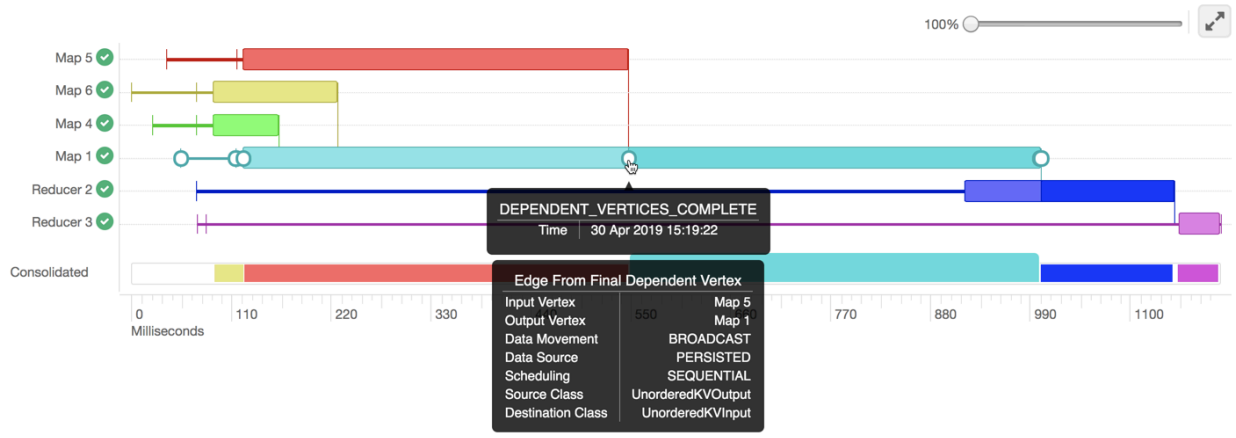


The following details can help you to view the time taken by a particular task and debug the query:

- Vertex ID: It is a unique identifier for a particular vertex.
- Status: Indicates whether the query executed successfully or not.
- Progress: Indicates the progress of the vertex, usually for long-running queries.
- Start time: Indicates when a particular vertex started.
- End time: Indicates when the particular vertex ended.
- Duration (in milliseconds): Indicates the total time taken by the vertex to complete its execution.
- First task start time: Indicates when the first task within that vertex started its execution.
- Tasks: Indicates the total number to tasks executed in a particular vertex.
- Succeeded tasks: Indicates the number of tasks that were executed successfully within that vertex.
- Running tasks: Indicates the tasks that are still running.
- Pending tasks: Indicates the tasks that have not yet started their execution.
- Processor class: It is the Hive processor for Tez that forms the vertices in Tez and processes the data. For example, org.apache.hadoop.hive.ql.exec.tez.ReduceTezProcessor, org.apache.hadoop.hive.ql.exec.tez.MapTezProcessor.

The vertical lines connecting two vertices denote the dependency of a vertex on another vertex, as shown in the following image:

Figure 3: DAG swimlane showing dependent vertices

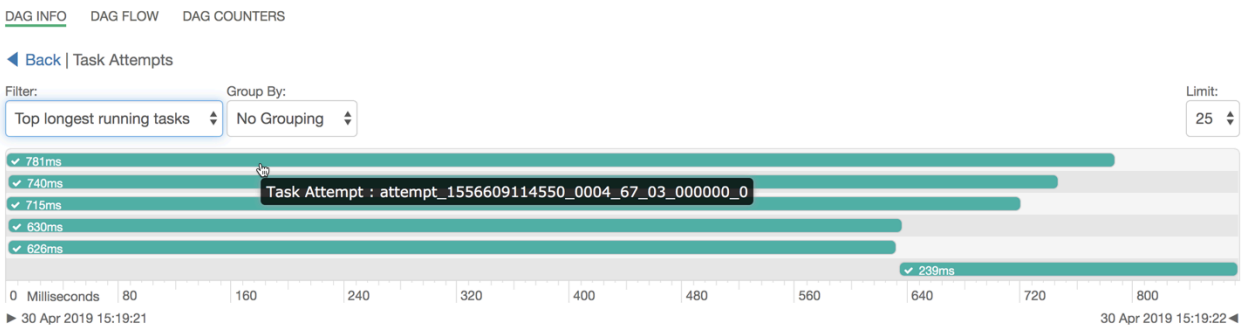


In the above example, Map 1 depends on the results of Map 5. Map 1 will finish its execution only when Map 5 finishes its execution successfully. Similarly, Reducer 2 depends on Map 1 to complete its execution.

The consolidated timeline shows the percentage of time each vertex took to complete executing. You can increase or decrease the scale of the timeline axis by moving the slider on the top right corner of the **DAG Info** section.

To further debug, click anywhere on the horizontal bar. This takes you to the **Task Attempts** section. The **Task Attempts** section shows the number of times a certain task was attempted. Each task attempt has a unique task attempt ID, as shown in the following image:

Figure 4: DAG info: Viewing task attempt



You can filter the result set by:

- Top longest running tasks: Used to filter the task that took the most time to complete.
- Errored tasks: Used to filter tasks that stopped because of some error.
- Tasks which started last: Used to filter the tasks which started late.

You can also group the result set either by tasks, containers in which the tasks were running, or the nodes on which the tasks were running.

For a query has more than one DAG ID associated with it, you can view its DAG by selecting the DAG ID from the dropdown.

DAS also allows you to download detailed logs of a task attempt. The log provides information about the container in which the task was running. To download the logs, click anywhere on the task attempt. On the **Details** pop-up, click Open log in new tab.

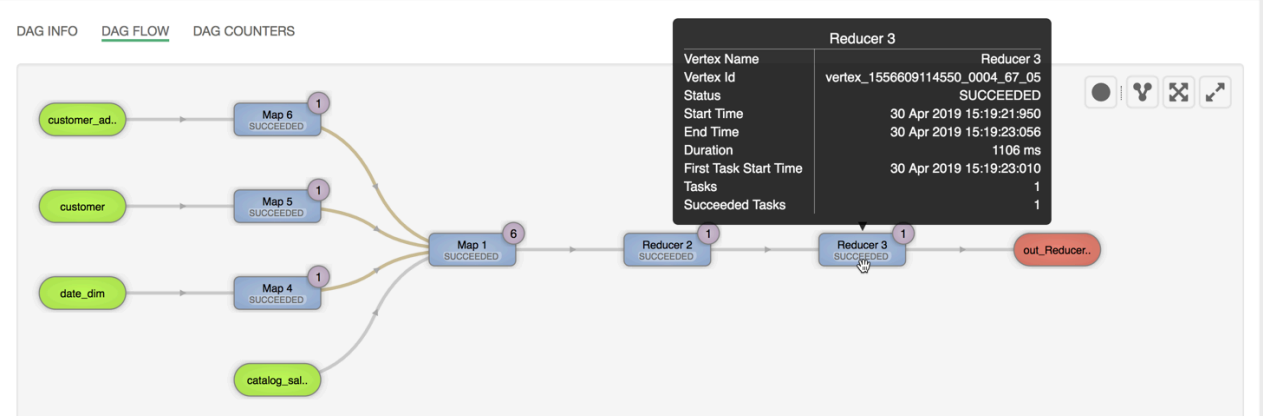
Note: Currently, the task-level logs are disabled for LLAP.

Viewing the DAG flow

The **DAG Flow** tab displays the Directed Acyclic Graph that is created by the Hive engine in the form of a flowchart.

The following image shows the DAG flow in the form of a flowchart:

Figure 5: DAG flowchart



Here, the inputs to vertices Map 4, Map 5, Map 6, and Map 1 are the tables displayed in green boxes. Next, Map 1 depends on the result set generated by Map 4, Map 5, and Map 6. Map 1 then passes its result as an input to Reducer 2. When the Reducer 2 finishes its execution, the results are passed on to Reducer 3. Reducer 3 is the last vertex in the DAG flow. After the Reducer 3 successfully completes its execution, the query output is written to a file in HDFS.

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 icon. You can switch between the horizontal and vertical orientation by clicking the Toggle orientation icon.

Viewing the DAG counters

The DAG counters provide 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 DAG counters provide details, such as:

- Number of bytes read and written
- Number of tasks that initiated and ran successfully
- Amount of CPU and memory consumed

Viewing the Tez configurations for a query

The DAG Configurations tab in the **DAG Info** section on the **Query Details** page provides the Tez configuration details for the query that has a DAG associated with it. You can search for a particular configuration by entering the configuration name in the search field.

Manage databases and tables

DAS enables you to completely manage and administer databases and tables. You can create and drop databases and tables, manage columns and partitions, edit tables, view storage information, view detailed information about the databse and the tables within it, view table and column statistics, and preview tables.

Using the Database Explorer

From the Database Explorer, you can select a database to manage it, create a new database, or drop a database.

Once you select a database, the tables list gets refreshed to list all the tables from the selected database.

You can perform the following actions using the Database Explorer:

- Select a database
- Create a database
- Drop a database
- Search for a table within a database
- Create a table

Searching tables

After selecting a database, the tables tab is refreshed to list all the tables in the database.

To search for a table, start by entering the name of the table in the Search box. The list gets refreshed as you enter your search string to narrow the list down to your desired table.

Click the Refresh icon to refresh the list of tables.

Managing tables

Using the Database Explorer UI, you can view information about table columns, partitions, storage, and metadata, as well as preview a few rows from the table. This enables you to ensure that your table contains exactly the information that you intend it to.

For each table, the following tabs provide respective details:

- Columns: You can view the details of each column of the table. You can also search for columns using the Search box.
- Partitions: You can view the details of columns that are partitions in the table. You can also search from the list of columns.
- Storage Information: You can view the storage information such as input format, output format, if the table is compressed, number of buckets, buck columns, and so on.
- Detailed Information: You can view details such as the name of the database, the owner of the table, the created time, the last accessed time, table type, and so on.
- Statistics: You can view the table statistics and column statistics.
- Data Preview: You can preview a few rows from the table.

Creating tables

You can create a new table or upload an existing table to add it to a particular database.

Procedure

1. On the Database Explorer tab, click the + icon in the Tables section.

   The **Create Table** screen appears.
2. Enter the name of the table.
3. Click Add New Column to add columns to the table.
4. For each column, specify the following detail:
   a) Enter the name of the column.
   b) Select the data type of the column from the drop down list.
   c) Click Advanced to add advanced details of each column.
5. Click Advanced tab of the **Create Table** page.
   a) Select Transactional if you want the table to be a transactional table.
   b) Click Add Location if you want to specify a location for the table.
   c) Select a file format from the drop down list of Add File Format section.
   d) Click Add Row Format to specify details for the rows.
6. Click Properties to add key-value properties of the table.
7. Click Create.

Uploading tables

You can upload existing tables in CSV, JSON, or XML format to create new tables in the database. You can use the uploaded tables as other tables, run queries, and generate reports on them like other tables.

Procedure

1. Click Upload Table.
2. Enter the file format details in the Select File Format section.
   a) File type: Select the file type from the drop down list.
   b) Field Delimiter: This appears if you select CSV as the file format. Enter a delimiter character such as comma.

      Note: Pick a delimiter that is not part of the column data.

   c) Escape Character: This appears if you select CSV as the file format. Enter an escape character such as \.
   d) Quote Character: This appears if you select CSV as the file format. Enter a quote character.
   e) If the first row is header, click to select Is first row header checkbox.
   f) If the file contains endlines, click to select Contains endlines checkbox.
3. Select the source of the table file.
   a) Upload from HDFS: Select this option to upload files from HDFS.
   b) Upload from Local: Select this option to upload files from the local node.

      Note: Uploading tables from Amazon S3 is not supported in the CDP 1.2 release of Cloudera Runtime.

   The table details automatically appear as preview. Click Preview if the table details do not load automatically. DAS updates the name of the table, names of the columns, and the data types of each column.
4. Modify the names of the table and columns as needed.
5. Click the Advanced tab to specify the following additional options:
   a) Select Transactional if you want the table to be a transactional table.
   b) Click Add Location if you want to specify a location for the table.
   c) Select a file format from Add File Format.
   d) Click Add Row Format to specify details for the rows.

6. Click Properties to add key-value properties of the table.
7. Click Create.

Editing tables

You can edit tables to add new columns, update existing columns, and make changes to the properties of the table.

Procedure

1. On the **Database Explorer** tab, select a database.
2. From the list of tables, select the table that you want to edit.
3. From the Table view, click the Actions menu.
4. Click Edit.
5. Edit the details of existing columns.
6. Click Add New Column to add columns to the table.
7. For each column, specify the following details:
   a) Enter the name of the column.
   b) Select the data type of the column from the drop down list.
   c) Click Advanced to add more details for each column.
8. Click the Advanced tab on the **Create Table** page to specify the following additional options:
   a) Select Transactional if you want the table to be a transactional table.
   b) Click Add Location if you want to specify a location for the table.
   c) Select a file format from the drop down list of Add File Format section.
   d) Click Add Row Format to specify details for the rows.
9. Click Properties to add or edit key-value properties of the table.
10. Click Edit to save the changes you made to the table.

Deleting tables

You can delete a table using the Database Explorer.

Procedure

1. On the Database Explorer tab, select the database.
2. From the list of tables, select the table that you need to delete.
3. From the Table view, click the Actions menu.
4. Click Delete.
5. In the dialog box that appears, click Confirm.

Managing columns

Using the Columns tab, you can view the details of each column of the table.

You can also search for columns using the Search box.

To search for a column, enter the name of the column in the search box and click Regex Search.

Managing partitions

On the Partitions tab, you can view the details of columns that are partitions in the table.

You can also search from the list of columns.

To search for a column that is a partition, enter the name of the column in the search box and click Regex Search.

Viewing storage information

Using the Storage Information tab, you can view the storage information of the table such as input format, output format, if the table is compressed, number of buckets, number of columns, and more details.

To search for particular details, enter the name in the search box and click Regex Search.

Viewing detailed information

Using the Detailed Information tab, you can view details such as the name of the database, the owner of the table, the created time, the last accessed time, table type, and more details.

To search for particular details, enter the name in the search box and click Regex Search.

Viewing table and column statistics

On the Statistics tab, you can view table statistics and column statistics.

To view the statistics of each column, click Show. The following details are displayed:

- Min
- Max
- Number of Nulls
- Distinct Count
- Average Column Length
- Max Column Length
- Number of True
- Number of False

If a message appears that the statistics might be stale, select Include Columns and click Recompute at the top of the page.

Previewing tables using Data Preview

On the Data Preview tab, you can preview a few rows from the table.

Manage reports

As a database administrator, you can view which columns and tables are used for joins and make changes to the data layout to optimize the performance of the query.

Note: The view reports feature is not available in the DAS-Lite version.

Viewing the Read and Write report

The Read and Write report displays a list of tables belonging to the selected database along with the read count, write count, and other details for each table.

To view the Read and Write report, click Reports Read and Write Report . The **Relations** page is displayed.

The left side lists the tables within the selected database along with the read count, write count, and so on. The right side shows these details in the form of an entity relationship diagram.

You can switch between the databases using the Database dropdown menu.

To further refine your search based on time, select the time period from the dropdown. After you select the time period, the corresponding “to” and “from” dates representing the selection are displayed along with the time zone. The time zone that is displayed is that of the DAS server.

For each table, select one of the following counts from the drop down list of counts:

- Projection Count
- Aggregation Count
- Filter Count
- Join Count

Viewing the Join report

The Join report provides you the details of joins between all the tables in a database.

To view the join report, click Reports Join Report . You can change the database from the dropdown list to view the join diagram for that database.

You can switch between the databases using the Database dropdown menu.

To refine your search based on time, select the time period from the dropdown. After you select the time period, the corresponding “to” and “from” dates representing the selection are displayed along with the time zone. The time zone that is displayed is that of the DAS server.

You can further refine your join diagram using the following join algorithms:

- Join
- Hash Join
- Merge Join
- Lateral View Join
- Sorted Merge

The diagram gets refreshed based on your selection.

If you hover over a particular column, then all the joins with the selected column are highlighted. If you hover over a particular connection line, then the number of times the two columns were joined (also known as the join count) is displayed. The join count that is displayed is for the time period that you select.

For every table present on the Join report, DAS displays the database in which that table belongs. If you select a particular database from the dropdown menu, then the Join report predominantly displays all the joins originating from the tables within the selected database to the tables belonging to other databases.

DAS administration using Cloudera Manager in CDP

The topics listed in this section describe how you can run a query on a different Hive instance, modify session cookie timeout period, configure user authentication using SPNEGO and LDAP, manually configure TLS/SSL, and clean up old queries and DAG information in a CDP Private Cloud Base cluster that is managed using Cloudera Manager.

Running a query on a different Hive instance

By default, ZooKeeper redirects your query to run on a Hive instance which is available at that time. However, if you have optimized a certain Hive instance, and you want to run your workload on that particular Hive instance, then you can select that Hive service to run your queries using Cloudera Manager.

About this task

Note: The Hive instance that you want to switch to should be within the same cluster.

Modifying the session cookie timeout value

By default, the session cookie expire in 86400 seconds (or 24 hours). Depending on your organizational requirement, you can increase or decrease the session cookie timeout duration from Cloudera Manager.

Before you begin

Note: The steps provided in this topic are applicable to DAS deployed on a CDP Private Cloud Base cluster that is managed using Cloudera Manager.

Procedure

1. From Cloudera Manager, go to Clusters DAS Configurations and specify the timeout duration in the Webapp Session Timeout field.
2. Save the changes and restart the required services by clicking Action Restart .

Configuring user authentication

You can authenticate the users by using either SPNEGO or LDAP, or choose not to set up user authentication.

If you do not want to enable user authentication, then you can select NONE from the DAS User Authentication field in Cloudera Manager.

Warning: If you select NONE, then all the queries will be executed as the hive service user.

Configuring user authentication using SPNEGO

SPNEGO uses a Kerberized environment for user authentication.

To use SPNEGO to authenticate users, select SPNEGO from the DAS User Authentication field in Cloudera Manager.

Configuring user authentication using LDAP

LDAP authenticates users using a directory server, such as MS Active Directory, OpenLDAP, or OpenDJ.

To use LDAP to authenticate users, select LDAP from the DAS User Authentication field in Cloudera Manager, and complete the rest of the configuration by entering details in the following fields:

- LDAP URL: Specify the LDAP URL to use for authentication.
- LDAP GUID Key: Specify the LDAP attribute name whose values are unique on this LDAP server. For example: uid or CN.
- LDAP Group Class Key: Specify the LDAP objectClass that each of the groups implements in LDAP. For example: objectClass: group.

- LDAP Group Membership Key: Specify the LDAP attribute name on the group object that contains the list of distinguished names for the user, group, and contact objects that are members of the group. For example: member, uniqueMember, or memberUid.
- LDAP User Membership Key: Specify the LDAP attribute name on the user object that contains groups of which the user is a direct member, except for the primary group, which is represented by the primaryGroupId. For example: memberOf.
- LDAP Basedn: Specify the base domain name.
- LDAP User DN Pattern: Specify a pattern for the distinguishedName (DN) for all the users in the directory. This value could be a single DN if the LDAP user entities are co-located within a single root, or it could be a colon-separated list of all the DN patterns if the users are scattered across different trees or forests in the directory.

  Each DN pattern can contain a %s in it that will be substituted with the username (from the user filter) by the provider for user search queries.

  For example: For single DN, you can specify CN=%s,CN=Users,DC=apache,DC=org. For two DNs, you can specify CN=%s,OU=Users,DC=apache,DC=org:uid=%s,CN=UnixUsers,DC=apache,DC=org.
- LDAP Group DN Pattern: Specify a pattern for the distinguishedName (DN) for all the groups in the directory. This value could be a single DN if the LDAP Group entities are co-located, or it could be a colon-separated list of all DN patterns if the groups are scattered across different trees.

  Each DN pattern can contain a %s in it that will be substituted with the group name (from the group filter) by the provider for group search queries.

  For example: For single DN, you can specify CN=%s,OU=Groups,DC=apache,DC=org. For two DNs, you can specify CN=%s,OU=Groups,DC=apache,DC=org:uid=%s,CN=Users,DC=apache,DC=org.
- LDAP Custom Query: There are several LDAP implementations available for commercial use, with no standard set of attributes within each implementation. If any of the filters listed in this topic do not meet the requirements for some unforeseen reasons, then DAS can use a user-specified LDAP query string to execute against the LDAP server. This configured query is expected to return a set of DNs that represent individual users. The returned result is then used to adjudicate a GRANT/DENY decision to the authenticating user. To support this configuration, this new configuration property has been introduced. For example: group1,group2.
- LDAP Group Filter: Specify the group name filter that is to be enforced by the LDAPAtnProvider. You can specify the individual groups using a comma-separated list. The user MUST belong to one or more of these groups for the authentication request to succeed. For example: group1,group2.
- LDAP User Filter: Specify a comma-separated list of all the usernames to whom you want to grant access. The Atn provider grants access to a user if the user is a part of this list, and denies access otherwise.
- LDAP Domain: Specify the base domain name.

Configuring TLS/SSL encryption manually for DAS using Cloudera Manager

To secure the transfer of sensitive information between Data Analytics Studio (DAS) and Cloudera Manager as well as with other services within your cluster, you must enable TLS/SSL for both Event Processor and WebApp. You can configure TLS manually using Cloudera Manager or have it set up automatically using the “Auto-TLS” feature.

About this task

If you have Auto-TLS enabled at the cluster-level, then DAS is also configured to use the TLS encryption. To check whether Auto-TLS is enabled for your cluster:

1. Sign in to Cloudera Manager as an administrator.
2. Go to Administration Security .
3. If Auto-TLS is enabled, the **Status** page displays “Auto-TLS is Enabled.”

   If Auto-TLS is not enabled, then click Enable Auto-TLS and complete the wizard.

If the TLS certificates are issued by your existing Certificate Authority (CA) to maintain a centralized chain of trust, then you can manually configure TLS for DAS. To manually configure TLS/SSL for DAS:

Procedure

1. Sign in to Cloudera Manager as an administrator.
2. Go to Clusters $Data Analytics Studio service Configuration and select Enable TLS/SSL for Data Analytics Studio Eventprocessor and Enable TLS/SSL for Data Analytics Studio Webapp Server.
3. Specify the location of the keystore and the truststore files and the corresponding passwords for both DAS Event Processor and WebApp server as explained in the following table:

Property	Description
Data Analytics Studio Eventprocessor TLS/SSL Server JKS Keystore File Location	Enter the location of the keystore file for the DAS Event Processor. This is used when the Event Processor is the server in a TLS/SSL connection. The keystore must be in JKS format.
Data Analytics Studio Eventprocessor TLS/SSL Server JKS Keystore File Password	Enter the password for the Event Processor JKS keystore file.
Data Analytics Studio Webapp Server TLS/SSL Server JKS Keystore File Location	Enter the location of the keystore file for the DAS WebApp. This is used when the WebApp is the server in a TLS/SSL connection. The keystore must be in JKS format.
Data Analytics Studio Webapp Server TLS/SSL Server JKS Keystore File Password	Enter the password for the WebApp JKS keystore file.
Data Analytics Studio Eventprocessor TLS/SSL Client Trust Store File	Enter the location of the truststore for the DAS Event Processor in JKS format. This is used when the Event Processor is the client in a TLS/SSL connection. The truststore must contain the certificate(s) used to sign the service(s) connected to. If this parameter is not provided, the default list of known certificate authorities is used instead.
Data Analytics Studio Eventprocessor TLS/SSL Client Trust Store Password	Enter the password for the Event Processor TLS/SSL Certificate Trust Store File. This password is not mandatory to access the truststore. If set, it checks the integrity of the file. The contents of truststores are certificates, and certificates are public information.
Data Analytics Studio Webapp Server TLS/SSL Client Trust Store File	Enter the location of the truststore for the DAS WebApp in JKS format. This is used when the WebApp server is the client in a TLS/SSL connection. The truststore must contain the certificate(s) used to sign the service(s) connected to. If this parameter is not provided, the default list of known certificate authorities is used instead.
Data Analytics Studio Webapp Server TLS/SSL Client Trust Store Password	Enter the password for the WebApp server TLS/SSL Certificate Trust Store File. This password is not mandatory to access the truststore. If set, it checks

Property	Description
	the integrity of the file. The contents of truststores are certificates, and certificates are public information.

4. Click Save Changes.
5. Restart the DAS service.

Cleaning up old queries, DAG information, and reports data

The DAS Postgres database stores all the queries that you run from the DAS UI or beeline, and all the data that is used to generate the DAG information and reports. Over a period of time, this can grow in size. To optimize the available capacity, DAS has a cleanup mechanism that, by default, purges all the queries and DAG information older than 30 days and purges old reports after 365 days. However, you can customize the cleanup frequency from Cloudera Manager.

About this task

To customize the cleanup intervals:

Procedure

1. Sign in to Cloudera Manager as an Administrator.
2. Go to Clusters Data Analytics Studio service Configuration and search eventprocessor_extra.properties.

   The Data Analytics Studio Eventprocessor Advanced Configuration Snippet (Safety Valve) for conf/props/eventprocessor_extra.properties field is displayed and is used to set or override properties for DAS in the form of key value pairs.
3. Add the cleanup.query-info.interval, cleanup.report-info.interval, and cleanup.cron.expression configurations in the event-processing section as shown in the following example:

```
event-processing= {
        hive.hook.proto.base-directory= "{{data_analytics_studio_event_p
rocessor_hive_base_dir}}",
        tez.history.logging.proto-base-dir= "{{data_analytics_studio_even
t_processor_tez_base_dir}}",
        meta.info.sync.service.delay.millis= 5000,
        actor.initialization.delay.millis= 20000,
        close.folder.delay.millis= 600000,
        reread.event.max.retries= -1,
        reporting.scheduler.initial.delay.millis= 30000,
        reporting.scheduler.interval.delay.millis= 300000,
        reporting.scheduler.weekly.initial.delay.millis= 60000,
        reporting.scheduler.weekly.interval.delay.millis= 600000,
        reporting.scheduler.monthly.initial.delay.millis= 90000,
        reporting.scheduler.monthly.interval.delay.millis= 900000,
        reporting.scheduler.quarterly.initial.delay.millis= 120000,
        reporting.scheduler.quarterly.interval.delay.millis= 1200000,
        cleanup.query-info.interval= 2592000,
        cleanup.report-info.interval= 31536000,
        cleanup.cron.expression= "0 0 2 * * ?"
  },
```

   In this example, the query data will be cleaned up after 2592000 seconds (which is equal to 30 days), the report data will be cleaned up after 31536000 seconds (which is equal to 365 days), and the cleanup jobs will be triggered to run at 02:00:00 hours (or 2 AM), as per the server timezone.
4. Click Save Changes.
5. Restart the DAS service.

DAS administration using Ambari in CDP

The topics listed in this section describe how you can run a query on a different Hive instance, clean up DAG data from the Postgres database, create system tables to run queries on DAG events, and change the retention period of DAS event logs in a CDP Private Cloud Base cluster that is managed using Ambari.

This section applies only if you have installed DAS on a CDP Private Cloud Base cluster using Ambari.

Running a query on a different Hive instance

By default, ZooKeeper redirects your query to run on a Hive instance that is available at that time. However, if you have optimized a certain Hive instance, and you want to run your workload on that particular Hive instance, then you can switch to that instance by specifying the HiveServer2 host name and the port in the JDBC connection field on the DAS user interface.

Before you begin

Note: The steps provided in this topic are applicable to DAS deployed on a CDP Private Cloud Base cluster that is managed using Ambari.

About this task

The JDBC connection string is defined by the hive.zookeeper.quorum property under Ambari Services Hive CONFIGS ADVANCED Advanced hive-site . It also resides in following DAS configuration file: etc/das/conf/das-hive-site.conf.

Note: The JDBC connection string that you specify through the web interface is a temporary configuration and is limited to that particular browser session.

To update the JDBC connection string:

Procedure

1. Sign in to the DAS portal.
2. Click the profile icon on the top right corner and then click About.
3. On the **Configurations** window, click Edit under the JDBC connection field.
4. Specify the HiveServer2 host name and port that you want to connect to in the following format:

   ```
   jdbc:hive2//HiveServer2-host:port/
   ```

   Note: The Hive instance that you want to switch to should be within the same cluster.
5. To save the changes, click Update.
6. To revert any changes that you have made to the original connection string, click Reset.
7. To go back to the **Configurations** window, click Cancel.
8. To exit, click OK.

Results

When you run your query after updating the JDBC connection string, the query is executed on that particular Hive instance instead of being redirected to any Hive instance by ZooKeeper.

Cleaning up old queries, DAG information, and reports data using Ambari

The DAS Postgres database stores all the queries that you run from the DAS UI or beeline, and all the data that is used to generate the DAG information and reports. Over a period of time, this can grow in size. To optimize the available capacity, DAS has a cleanup mechanism that, by default, purges all the queries and DAG information older than 30 days and purges old reports after 365 days. However, you can customize the cleanup frequency by adding the cleanup.query-info.interval, and cleanup.report-info.interval configurations, and the Cron expression: cleanup.cron .expression in the das-event-processor.json file from Ambari.

About this task

Note:

The steps provided in this topic are applicable to DAS deployed on a CDP Private Cloud Base cluster that is managed using Ambari.

Procedure

1. From the Ambari UI, go to Data Analytics Studio CONFIGS Advanced data_analytics_studio-event_processor-properties .
2. To customize the cleanup intervals, under Data Analytics Studio Event Processor config file template, add the three new configurations in the event-processing section as shown in the following example:

```
"event-processing": {
        "hive.hook.proto.base-directory": "{{data_analytics_studio_event_p
rocessor_hive_base_dir}}",
        "tez.history.logging.proto-base-dir": "{{data_analytics_studio_ev
ent_processor_tez_base_dir}}",
        "meta.info.sync.service.delay.millis": 5000,
        "actor.initialization.delay.millis": 20000,
        "close.folder.delay.millis": 600000,
        "reread.event.max.retries": -1,
        "reporting.scheduler.initial.delay.millis": 30000,
        "reporting.scheduler.interval.delay.millis": 300000,
        "reporting.scheduler.weekly.initial.delay.millis": 60000,
        "reporting.scheduler.weekly.interval.delay.millis": 600000,
        "reporting.scheduler.monthly.initial.delay.millis": 90000,
        "reporting.scheduler.monthly.interval.delay.millis": 900000,
        "reporting.scheduler.quarterly.initial.delay.millis": 120000,
        "reporting.scheduler.quarterly.interval.delay.millis": 1200000,
        “cleanup.query-info.interval”: 2592000,
        “cleanup.report-info.interval”: 31536000,
        “cleanup.cron.expression: "0 0 2 * * ?"
  },
```

   In this example, the query data will be cleaned up after 2592000 seconds (which is equal to 30 days), the report data will be cleaned up after 31536000 seconds (which is equal to 365 days), and the clean up jobs will be triggered to run at 02:00:00 hours (or 2 AM), as per the server timezone.
3. Click Save.
4. Restart all the required services.

Creating system tables to run query on Hive and Tez DAG events

The Tez DAG events are exposed in the dag_data table, and the Hive query events are exposed in the query_data table in the sys database. You can run Hive queries from DAS to process these events and generate custom reports.

Before you begin

Note: The steps provided in this topic are applicable to DAS deployed on a CDP Private Cloud Base cluster that is managed using Ambari.

About this task

You need to create these system tables before you can run queries on them.

Procedure

1. From Ambari, go to Data Analytics Studio.
2. Click Actions Create Tables .
3. Navigate through the table creation wizard and make sure that the operation completes successfully.

   You can view the dag_data and the query_data tables under the sys database in the DAS UI.

Changing the retention period of DAS event logs

If you click on a task on the DAG swimlane for a query for which the event logs were cleaned up, you see the “DAG info not available” message on the task attempt page. You can increase the retention period to resolve this issue.

About this task

By default, the event logs and the DAG data is retained for seven days, after which the DAG history is purged from the system. This is defined in the hive.hook.proto.events.ttl parameter.

To change the retention period of the event logs and to view task-level DAG information for older queries:

Procedure

1. From Ambari, go to Hive HiveMetaStore Configs .
2. Specify the time to live in the hive.hook.proto.events.ttl parameter.

   For example, to retain the logs for 30 days and view task-level DAG information for queries that were run 30 days ago, specify 30d.
3. Save the changes and restart the DAS service.