Cloudera Data Warehouse Private Cloud 1.5.0

Managing Virtual Warehouses

Date published: 2020-08-17
Date modified: 2023-01-25

https://docs.cloudera.com/

Legal Notice

Contents

Adding a new Database Catalog

In addition to the default Database Catalog, created automatically, you can add additional Database Catalogs if you want a standalone data warehouse that is not shared with other authorized users of the environment.

About this task

When you activate an environment from the Data Warehouse, a default Database Catalog is created and named after your environment. This HMS instance associated with the default Database Catalog is the same HMS as the one used by your CDP environment. You can add additional Database Catalogs if you want standalone data warehouses based on a new HMS instance. When you create a new Database Catalog, you specify which environment to use. If you make a change to the default database catalog, the change is reflected in the environment where the default Database Catalog resides. However, if you make any change to the non-default database catalogs, the change is not reflected in that environment.

You can optionally load demo data in Hue when you create a new Database Catalog.

Before you begin

You need to obtain the DWAdmin role.

Procedure

1. Log in to the CDP web interface, navigate to Data Warehouse Database Catalogs Add New .
2. In Name, specify a Database Catalog name.

   Note: CDW uses deterministic namespace and adds a prefix to the Database Catalog name. The length of the namespace ID after CDW applies a prefix to the Database Catalog name, including the hyphen (-), should not exceed 63 characters. You can specify the Database Catalog name 53 characters long.
3. In Environments, select the name of an activated environment.
4. For example purposes, turn on Load Demo Data to use sample airline data in Hue.
5. Click Create to create the new Database Catalog.

Adding a new Virtual Warehouse

This topic describes how to create a Virtual Warehouse in Cloudera Data Warehouse (CDW) Private Cloud service.

About this task

In CDW service, a Virtual Warehouse is an instance of compute resources that is equivalent to a cluster. A Virtual Warehouse provides access to the data in tables and views in the data lake that correlates to a specific Database Catalog. Virtual Warehouses can only lookup the Database Catalog that they have been configured to access.

Required role: DWAdmin

Before you begin

Before you create a new Virtual Warehouse, determine what is the number of concurrent queries or users your Virtual Warehouse must serve during peak periods. This information helps you determine what size of Virtual Warehouse you need. Choose the size based on the number of nodes you typically use for clusters in an on-premises deployment. Also consider the complexity of your queries and the size of the data sets that they access. Larger sized warehouses with more nodes can cache more data, which enhances performance.

Virtual Warehouse sizes you can choose from:

Virtual Warehouse Size	Number of Nodes
XSMALL	2
SMALL	10
MEDIUM	20
LARGE	40

Procedure

1. Log in to the CDP web interface and navigate to the Data Warehouse service.
2. In the Data Warehouse service, click Virtual Warehouses in the left navigation panel.
3. On the Virtual Warehouses page, click Add New.
4. In the New Virtual Warehouse dialog box, specify a Virtual Warehouse name, the Type (Hive or Impala), which Database Catalog it queries, and the size.
5. After you choose a size, you can configure auto-scaling settings.
6. Click Create to create the new Virtual Warehouse.

Configuring Impala coordinator high availability

A single Impala coordinator might not handle the number of concurrent queries you want to run or provide the memory your queries require. You can configure multiple active coordinators to resolve or mitigate these problems. You can change the number of active coordinators later.

About this task

You can configure up to five active-active Impala coordinators to run in an Impala Virtual Warehouse. When you create an Impala Virtual Warehouse, CDW provides you an option to configure Impala coordinator and Database Catalog high availability, described in the next topic. You can choose one of the following options:

Disabled

Disables Impala coordinator and Database Catalog high availability

Active-passive

Runs multiple coordinators (one active, one passive) and Database Catalogs (one active, one passive)

Active-active

Runs multiple coordinators (both active) and Database Catalogs (one active, one passive)

By using two coordinators in an active-passive mode, one coordinator is active at a time. If one coordinator goes down, the passive coordinator becomes active.

If you select the Impala coordinators to be in an active-active mode, the client software uses a cookie to keep a virtual connection to a particular coordinator. When a coordinator disappears for some reason, perhaps due to a coordinator shutting down, then the client software may print the error "Invalid session id" before it automatically reconnects to a new coordinator.

Using active-active coordinators, you can have up to five coordinators running concurrently in active-active mode with a cookie-based load-balancing.

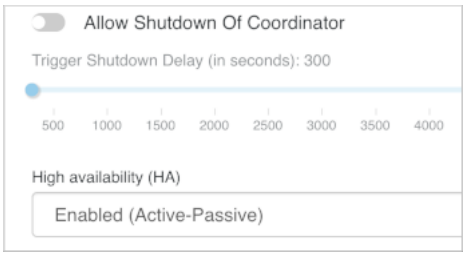
An Impala Web UI is available for each coordinator which you can use for troubleshooting purposes.

Clients who connect to your Impala Virtual Warehouse using multiple coordinators must use the latest Impala shell. The following procedure covers these tasks.

Procedure

1. Follow instructions for "Adding a new Virtual Warehouse".
2. Select the number of executors you need from the Size dropdown menu.

   A number of additional options are displayed, including High availability (HA).



3. Select the Enabled (Active-Active) option from the High availability (HA) dropdown menu.
4. Select the number of coordinators you need from the Number of Active Coordinators dropdown menu ranging from 2 to 5.

   You can edit an existing Impala Virtual Warehouse to change the number of active coordinators.

   Important: Do not decrease the number of active-active coordinators you set up initially; otherwise, the Virtual Warehouse may shut down immediately. If clients are running queries on the Virtual Warehouse, the queries could fail.
5. Change values for other settings as needed, click Create, and wait for the Impala Virtual Warehouse to be in the running state.

   Click ⓘ to learn more about the setting.
6. Go to Cloudera Data Warehouse Overview Impala Virtual Warehouse ⋮ Edit WEB UI , and then click each Impala Coordinator Web UI *n* link to get information about the coordinator.
7. Go to Cloudera Data Warehouse Overview Impala Virtual Warehouse ⋮ and select the Copy Impala shell Download command option.

   The following command is copied to your clipboard:

```
pip install impala-shell==4.1.0
```

8. Provide the command to clients who want to connect to the Impala Virtual Warehouse with multiple coordinators using the Impala shell.
9. Instruct the client user to update impyla to version compatible with CDW, as listed in Data Warehouse Release Notes in section, “Runtime component versions for Cloudera Data Warehouse Private Cloud”.

   For example, installing/updating impyla 0.18a2, is required to connect to your Virtual Warehouse active-active coordinators in CDW 2021.0.3-b27 or later.
10. Inform the client that to connect over ODBC to an HA-configured Impala Virtual Warehouse that uses active-active coordinators, you must append impala.session.id to the HTTPAuthCookies connector configuration option of the Cloudera ODBC driver.

Table 1: HTTPAuthCookies

Key Name	Value	Required
HTTPAuthCookies	impala.auth,JESSSIONID,KNOXSESSIONID,impala.session.id	Yes

Enabling Impala to spill to HDFS

When you create a new Impala Virtual Warehouse in Cloudera Data Warehouse Private Cloud, you can configure heavy Impala queries to write intermediate files during large sorts, joins, aggregations, or analytic function operations to a remote scratch space on HDFS.

Before you begin

Configure the Impala daemon to use the specified locations for writing the intermediate files as described in Configuring Impala daemon to spill to HDFS.

Note: You must create a new Impala Virtual Warehouse to enable the option to spill intermediate Impala query execution data to HDFS.

Procedure

1. Log in to the Data Warehouse service as a DWAdmin.
2. Click + under Virtual Warehouses on the **Overview** page to create a new Virtual Warehouse.
3. Specify a name for the Virtual Warehouse, select IMPALA as the type, select a Database Catalog, and size from the drop-down menu.
4. Specify the HDFS URI in the Spill to HDFS field in the following format:

   ```
   hdfs://[***HOSTNAME***]:[***PORT***]/[***PATH***]:[***LIMIT***]
   ```

   Hostname and port are mandatory arguments that you must specify in the HDFS URI.

   Note: When a valid HDFS URI is passed by the client, the 300G of local storage is used as a local disk buffer for spilling to HDFS.
5. Select scaling and resource allocation and click Create.

Configuring Impala coordinator shutdown

To optimize resource utilization, you need to know how to configure Impala coordinators to automatically shutdown during idle periods. You need to know how to prevent unnecessary restarts. Monitoring programs that periodically connect to Impala can cause unnecessary restarts.

About this task

When you create a Virtual Warehouse, you can configure Impala coordinators to automatically shutdown during idle periods. The coordinator start up can last several minutes, so clients connected to the Virtual Warehouse can time out.

Before you begin

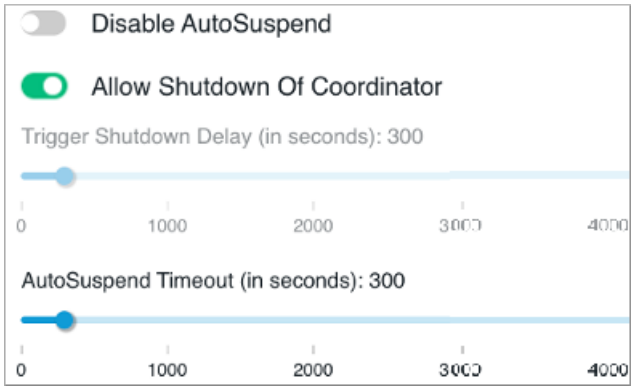
Update impyla, jdbc, impala shell clients if used to connect to Impala.

Procedure

1. Follow instructions for "Adding a new Virtual Warehouse".
2. Select a size for the Virtual Warehouse.
3. Turn off Disable AutoSuspend.

   The Impala coordinator does not automatically shutdown unless the Impala executors are suspended.

4. Turn on Allow Shutdown of Coordinator.



   After Impala executors have been suspended, the Impala coordinator waits for the time period specified by the Trigger Shutdown Delay before shutting down.

   For example, if AutoSuspend Timeout = 300 seconds and Trigger Shutdown Delay=150 seconds, after 300 seconds of inactivity Impala executors suspend, and then 150 seconds later, the Impala coordinator shuts down.

5. Accept default values for other settings, or change the values to suit your use case, and click CREATE.

   Click the tooltip ⓘ for information about a setting.

Correcting the Virtual Warehouse size

The size of the Virtual Warehouse you select during Virtual Warehouse creation determines the number of executors and concurrent queries the Virtual Warehouse can run. You need to know how to change the size of the Virtual Warehouse upward or downward to tune performance and manage cost.

About this task

You cannot change the size of a Virtual Warehouse, but you can handle incorrect sizing in the following ways.

- You can delete the Virtual Warehouse, and then recreate it in a different size.
- You can change the auto-scaling thresholds to change the effective size of the Virtual Warehouse based on demand. The actual size does not change, but increases or decreases in resources occurs automatically.

This task assumes you have two Virtual Warehouses that you decide are incorrectly sized for some reason. You correct the sizing of one by deleting and recreating the Virtual Warehouse. You correct the effective sizing of the other by changing auto-scaling thresholds.

Before you begin

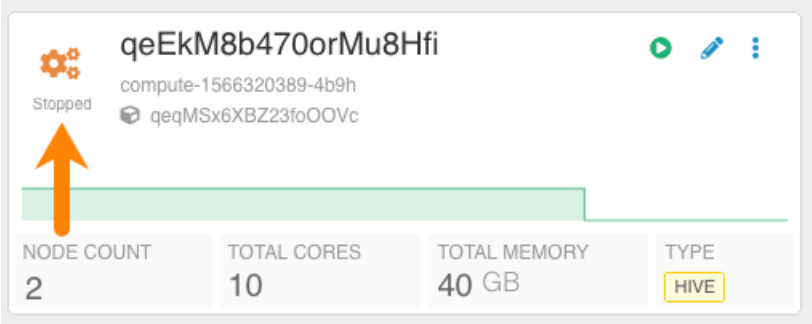
- You obtained the DWAdmin role.

Procedure

First Virtual Warehouse: Replace this Virtual Warehouse

1. Log in to the CDP web interface, navigate to Data Warehouse Overview , note the name of the Virtual Warehouse you want to modify, and note which Database Catalog it is configured to access.

2. In the Virtual Warehouse you want to delete, click Suspend to stop running it.



3. Click the options of the Virtual Warehouse you want to delete, and select Delete.
4. Click Virtual WarehousesAdd New.
5. Set up the new Virtual Warehouse:
   - Type the same Name for the new Virtual Warehouse as you used for the old Virtual Warehouse.

     Note: The fully qualified domain name of your Virtual Warehouse, which includes the Virtual Warehouse name plus the environment name must not exceed 64 characters; otherwise, Hue cannot load.
   - In Type, click the SQL engine you prefer: Hive or Impala.
   - Select your Database Catalog and User Group if you have been assigned a user group.
   - In Size, select the number of executors, for example xsmall-2Executors.
   - Accept default values for other settings, or change the values to suit your use case.

     Click for information about settings.
6. Click Create.

Second Virtual Warehouse: Change the Auto-Scaling Thresholds

7. In Data Warehouse Overview , click the options of the other Virtual Warehouse, a Hive Virtual Warehouse for example, to change auto-scaling thresholds, and select Edit.
8. In Sizing and Scaling, in Concurrency Autoscaling, slide the control to change the Max number of executors.
9. Click Apply Changes.

Auto-scaling Virtual Warehouses

This topic provides an overview of auto-scaling in Cloudera Data Warehouse (CDW) Private Cloud.

Virtual Warehouses can use Hive or Impala as the underlying execution engine. Typically, Hive is used to support complex reports and enterprise dashboards. Impala is used to support interactive, ad-hoc analysis. When you create a Virtual Warehouse, you set auto-scaling to make sure you have adequate resources to meet increases in demand. Auto-scaling settings also insure that your Virtual Warehouse relinquishes resources when demand decreases to save costs.

Auto-scaling: where scaling and concurrency meet

Scaling is the total capacity of the system and how elastic it is. System capacity requirements are based on the size of the largest query you need to run on a warehouse. *Concurrency* is the number of queries that can run at the same time in the same Virtual Warehouse.

In traditional deployments, scaling and concurrency must be planned for before you deploy your warehouse. In the cloud, the ability to acquire better scaling and concurrency elastically in response to workload demand enables

the system to operate more efficiently than the maximum limits you plan for. If you run your Virtual Warehouse configured to accommodate your peak workload as a constant default configuration, you might have inefficient resource utilization when system demand falls below that level.

Caching and auto-scaling

In CDW Private Cloud service frequently accessed data is cached on HDFS. This caching ensures that the data can be quickly retrieved for subsequent queries, boosting data warehouse performance.

Fault-tolerance and auto-scaling

Virtual Warehouses can tolerate single-node failures of any of its workers and can continue running active queries. Auto-scaling separates nodes from each other. This node separation provides better protection when a rogue query is submitted to the warehouse. In this scenario, node failures are limited to the auto-scaled nodes, which limits the impact of the rogue query to a small part of the Virtual Warehouse. The choice to have more auto-scaling groups indirectly allows for the system to tolerate such scenarios, so it is always recommended to use auto-scaling, even if the workload is a predictable one.

Tuning Hive Virtual Warehouses on private clouds

This topic describes how to tune Hive Virtual Warehouses in Cloudera Data Warehouse (CDW) Private Cloud.

About this task

When you tune Hive Virtual Warehouses, you set the auto-suspend timeout, the minimum and maximum number of nodes for your virtual cluster, when your cluster should scale up, and when it should scale down.

Procedure

1. Log in to the CDP web interface and navigate to the Data Warehouse service.
2. In the Data Warehouse service, click Overview in the left navigation pane.

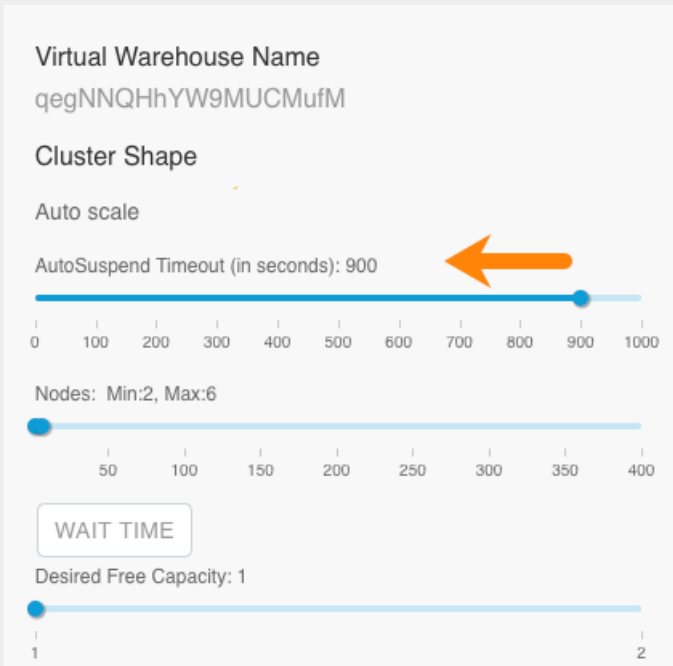
   

   Note: You can also tune your data warehouse on the Virtual Warehouse page using the same steps.
3. In the Overview page under Virtual Warehouses, click

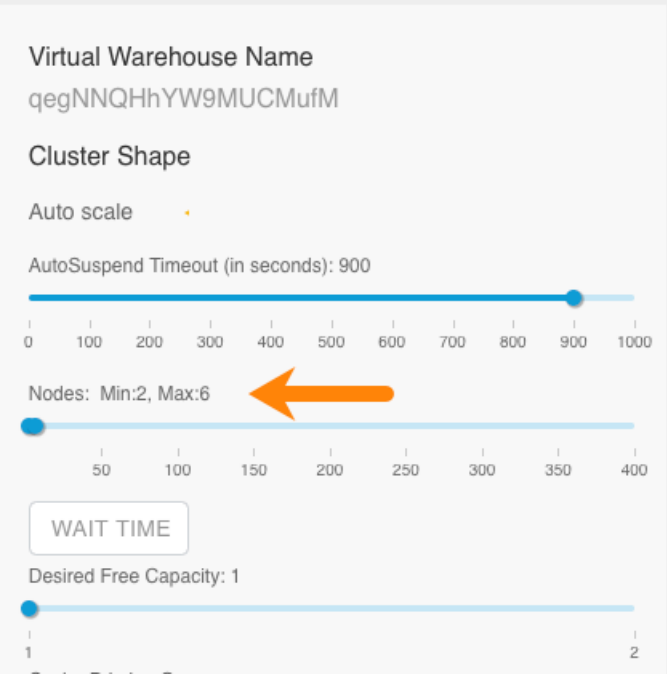
   

   against the required virtual warehouse and select Edit.

4. Click the SIZING AND SCALING tab to view the properties that you can adjust to tune auto-scaling for your data warehouse:
   a) Set the AutoSuspend Timeout property under Auto scale, which determines how many seconds the warehouse cluster is idle before it suspends itself:



   This setting helps to ensure performance is not impacted by having idle resources.
   b) Set the minimum and maximum number of nodes that the cluster can contain:



   Use the minimum number of nodes setting to ensure that your workloads always have resources and use the maximum number of nodes setting to contain having too many idle resources. Decide the minimum and maximum number of nodes based on your workloads similarly to how you determine node counts for your on-premises clusters. Consider the number of concurrent queries, the complexity of queries, and the volume

of queries in your workloads to determine the appropriate number of nodes to set on each Virtual Warehouse instance.

c) Choose when your cluster auto-scales up based on the WAIT TIME setting, which sets how long queries wait in the queue to run before the cluster auto-scales up. For example, if WaitTime Seconds is set to 10, then when executing queries are waiting in the queue for 10 seconds, the cluster auto-scales up to meet query demand.

Note: Scaling might react to non-scalable factors to spin up clusters. For example, query wait times might increase because of inefficient queries and not because of query volume.

d) Select Query Isolation if you have scan-heavy, data-intensive queries in your workloads.

Query Isolation enables your Virtual Warehouse to determine, based on the value you set for the hive.query.isolation.scan.size.threshold configuration parameter, whether to spawn dedicated nodes to run scan-heavy, data-intensive queries.

You can set this threshold parameter in the Virtual Warehouse details page for the warehouse:

1. In the Data Warehouse service UI, click Virtual Warehouses in the left navigation pane.
2. From the list of warehouses, click the Virtual Warehouse you want to configure this parameter for.
3. In the Virtual Warehouse details page, click CONFIGURATIONS Hiveserver2 .
4. Select hive-site from the Configuration files drop-down list and type isolation in the search text box to locate the parameter.
5. In the VALUE text box for the hive.query.isolation.scan.size.threshold parameter, enter the amount of data for your threshold in storage units. For example, 400GB.
6. Click APPLY to save your settings.

After you enable Query Isolation, two more configuration options appear:

- Max Concurrent Isolated Queries: Sets the maximum number of isolated queries that can run concurrently in their own dedicated executor nodes. Select this number based on the scan size of the data for your average scan-heavy, data-intensive query.
- Max Nodes Per Isolated Query: Sets how many executor nodes can be spawned for each isolated query.

e) Click APPLY.

Tuning Impala Virtual Warehouses

This topic describes how to tune Impala Virtual Warehouses in Cloudera Data Warehouse (CDW) Private Cloud.

About this task

When you tune Impala Virtual Warehouses, you can disable the auto-suspend feature, set the minimum and maximum executor nodes allocated for the warehouse and you can set the scale up and scale down delay which determines auto-scaling behavior.

Required role: DWAdmin

Procedure

1. Log in to the CDP web interface and navigate to the Data Warehouse service.
2. In the Data Warehouse service, navigate to the Overview page.

Note: You can also tune your warehouse on the Virtual Warehouse page using the same steps.

3. On the Overview page under Virtual Warehouses, click the edit icon for an Impala Virtual Warehouse in the upper right corner of the tile.

4. The next page provides properties that you can adjust to tune auto-suspend and auto-scaling for your Virtual Warehouse:
   a) Set the auto-suspend behavior:
      - If you do not want your Virtual Warehouse to auto-suspend, click Disable AutoSuspend. If you enable this feature, your Virtual Warehouse does not suspend itself, even if it is idle and no workloads are being processed.
      - If you do not want to disable auto-suspend, set the AutoSuspend Timeout. This sets the time, in seconds, that it takes for the Virtual Warehouse to automatically suspend itself. A Virtual Warehouse auto-suspends itself when the auto-scaler has scaled back to the last executor group and those executors are idle.
   b) Adjust the minimum or maximum number of executor nodes as needed:

      Setting and adjusting the minimum and maximum number of executor nodes per Virtual Warehouse is very similar to setting the number of nodes for on-premises clusters. Keep in mind the number of concurrent queries, the complexity of queries, and the volume of queries in your workloads to determine the appropriate number of executor nodes to set on each Virtual Warehouse instance.
   c) Set the Scale Up Delay and the Scale Down Delay to fine-tune when the auto-scaler starts scaling up and the number of executor groups to meet workload demand.
      - Scale Up Delay: Sets the length of time in seconds that the system waits before adding more executors when it detects queries waiting in the queue to execute. The time to auto-scale up is affected by how the underlying Kubernetes system is configured.
      - Scale Down Delay: Sets the length of time in seconds that the system waits before it removes executors when it detects idle executor groups. As with the Scale Up Delay setting, the time to auto-scale down is affected by how the underlying Kubernetes system is configured.
5. (Optional) If you need to tune your Impala Virtual Warehouse to run more queries per executor group, select Use Legacy Multithreading Mode.

   Note: By default Impala Virtual Warehouses can run 3 large queries per executor group. Executors can handle more queries that are simpler and that do not utilize concurrency on the executor. When you enable legacy multithreading, the Virtual Warehouse can run 12 queries per executor group. For most read-only queries the default setting of 3 queries per executor group is sufficient.
6. Click Apply in the upper right of the page to save your changes.

Auto-scale threshold settings

This topic provides information about the auto-scaling threshold settings for Hive and Impala Virtual Warehouses in Cloudera Data Warehouse (CDW) Private Cloud.

When you create new Virtual Warehouse instances, you can set auto-scaling thresholds. These thresholds set limits on automatic cluster scaling to meet workload demands. Setting these limits prevents warehouses from consuming too many resources when workload demands increase or decrease. Another important benefit of enabling auto-scaling for your Virtual Warehouse is that it further enforces node isolation, increasing warehouse fault tolerance. You can adjust the following auto-scaling thresholds:

Hive-LLAP Virtual Warehouse auto-scaling threshold settings

The following settings are available to configure auto-scaling for Hive-LLAP Virtual Warehouses:

Hive-LLAP Auto-scaling Threshold	Description
AutoSuspend Timeout	Sets the maximum time the warehouse idles before shutting down.

Hive-LLAP Auto-scaling Threshold	Description
Nodes: Min: <n>, Max: <n>	Sets the minimum and maximum number of nodes (executors) for the warehouse cluster. The maximum number of executors is limited by your cloud account limits. Choose the minimum and maximum number of executors based on two factors: • Average number of queries that must be run concurrently for your workloads. The more queries that must be run concurrently, the larger number of executors are needed. • The size of the data your workloads access. Larger numbers of executors can cache more data, which enhances performance.
WAIT TIME	Sets how long queries wait in the queue to execute. For example, if WaitTime Seconds is set to 10, then when executing queries are waiting in the queue for 10 seconds, the cluster auto-scales up to meet query demand.
Query Isolation	Enables the Virtual Warehouse to determine, based on the value you set for the hive.query.isolation.scan.size.threshold configuration parameter, whether to spawn dedicated executor nodes to execute scan-heavy, data-intensive queries in isolation.
Max Concurrent Isolated Queries	Available if Query Isolation is enabled. Sets the maximum number of isolated queries that can execute concurrently in their own dedicated executor nodes. Select this number based on the scan size of the data for your average scan-heavy, data-intensive query.
Max Nodes Per Isolated Query	Available if Query Isolation is enabled. Sets how many executor nodes can be spawned for each isolated query.

Impala Virtual Warehouse auto-scaling threshold settings

The following settings are available to configure auto-scaling for Impala Virtual Warehouses:

Impala Auto-scaling Setting	Description
Disable AutoSuspend	When you enable this control, your Virtual Warehouse does not suspend itself when the auto-scaler has scaled back to the last executor group, and those executors are idle. Instead, the Virtual Warehouse continues to consume cloud resources. You can override this behavior by disabling the Disable AutoSuspend control.
AutoSuspend Timeout	Sets the maximum time the warehouse idles before shutting down. This setting only applies when the Disable AutoSuspend control is not enabled.
Nodes: Min: <*n*> Max: <*n*>	Sets the minimum and maximum number of nodes (executors) for the warehouse cluster. The maximum number of executors is limited by your cloud account limits. Choose the minimum and maximum number of executors based on two factors: • Average number of queries that must be run concurrently for your workloads. The more queries that must be run concurrently, the larger number of executors is needed. • The size of the data your workloads access. Larger numbers of executors can cache more data, which enhances performance.
Scale Up Delay	Sets the length of time in seconds that the system waits before adding more executors when it detects queries waiting in the queue to execute. The time to auto-scale is affected by how the underlying Kubernetes system is configured.
Scale Down Delay	Sets the length of time in seconds that the system waits before it removes executors when it detects idle executor groups. As with the Scale Up Delay setting, the time to auto-scale down is affected by how the underlying Kubernetes system is configured.

Compaction in Cloudera Data Warehouse

You understand the importance of compaction and the consequences of neglecting to perform compaction. Compaction keeps your Data Warehouse healthy.

Over time tables belonging to a workload become fragmented due to operations performed on them by your workload users. These small, obsolete files might lead to performance degradation and query latency problems. Compaction

plays a major role in improving response time to workload queries by reducing the number of underlying files for a table and eliminating the obsolete ones. Compaction runs periodically in the background to maintain the optimal state.

Running periodic compaction is a best practice for the performance for ACID transactions. ACID inserts and deletes generate the problematic files that you might need to monitor and manage. In Cloudera Data Warehouse (CDW), compaction is always performed by a Hive Virtual Warehouse.

How compaction works

When data changes are made on Cloudera Data Warehouse (CDW) with inserts, updates, and deletes, delta files are created. The more changes that are made, the more delta files are created. When a large number of delta files are created, query performance degrades. Compaction removes these delta files to enhance query performance.

There are two types of compaction:

- Minor compaction: compacts multiple delta files into a single delta file.
- Major compaction: compacts one or more delta files and the base file for the bucket and creates a single new base file per bucket.

The goal of compaction is to "self heal" tables in order to restore the baseline query performance. All compactions are done in the background and do not prevent concurrent reads and writes of the data. After compacting, the system waits for all readers of the old files to finish and then removes the old files.

Compactor processes

These background processes run inside the metastore and HiveServer2 in Cloudera Data Warehouse (CDW) Private Cloud. They support the data modifications made as a result of ACID transactions.

Compactor process	Description
Initiator	This process runs in the metastore, which equates to the Database Catalog construct in the CDW UI, and discovers which tables and partitions are due for compaction. By default, it runs every 5 minutes.  **Important:** For the default Database Catalog, which is the Database Catalog created automatically when an environment is activated, all compaction takes place on CDP Base. To change this interval: **1.** Identify the Database Catalog for the Virtual Warehouse on which you want to change the compaction interval by selecting the Virtual Warehouse tile. The associated Database Catalog is highlighted. **2.** In the Database Catalog, click the edit icon in the tile to launch the Database Catalog details page. **3.** On the Database Catalog details page, make sure the CONFIGURATIONS tab is selected, and then select the Metastore subtab. **4.** On the Metastore subtab, select hive-site from the drop-down list on the left, and search for the hive .compactor.check.interval KEY. **5.** Add your preferred check interval in the associated VALUE field in seconds. **6.** Click APPLY in the upper right corner of the page to apply your changes. The services are automatically updated with the new configuration.
Worker	This process runs in HiveServer2, which equates to the Hive Virtual Warehouse construct in the CDW UI. The worker process performs the actual compacting work. In CDW, compaction runs an INSERT statement created from the output of a SELECT statement, thereby re-writing the data to new base or delta files.
Cleaner	This process runs in the metastore and deletes delta files after compaction and after it determines the files are no longer needed. By default, the cleaner runs every 5 seconds (5,000 milliseconds). The check occurs on the visibility ID/transaction ID, which is a global transaction identifier.

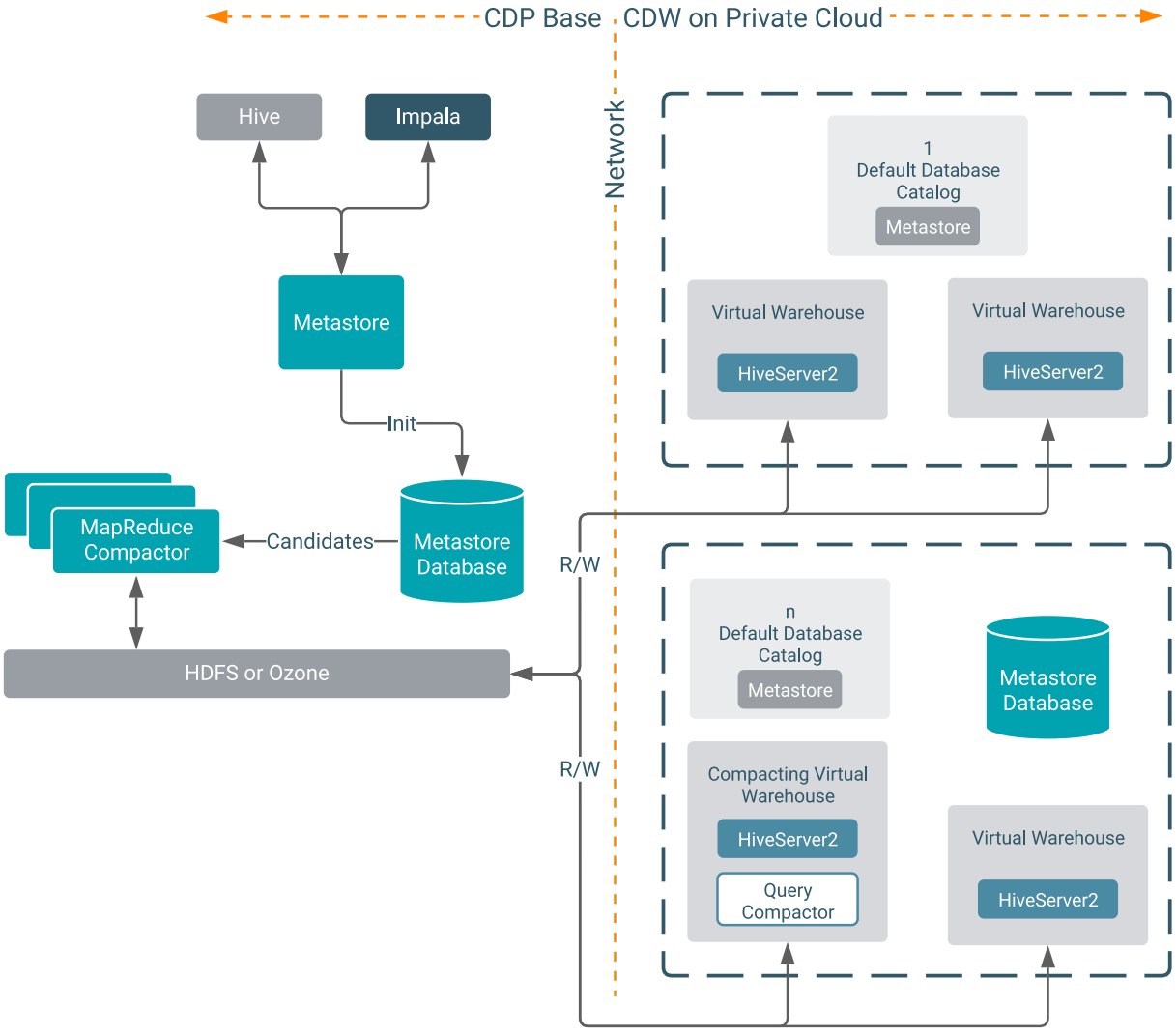
How compaction interacts with CDP Base

In CDP Base, the initiator and cleaner processes also run in the metastore as they do in Cloudera Data Warehouse (CDW) Private Cloud. However, the worker process runs in HiveServer2 as a MapReduce task so its progress can be viewed in YARN.

In CDW, the initiator and cleaner processes run in the Database Catalog, which is the CDW UI construct that equates to the metastore. The default Database Catalog, which is created by the system when you activate an environment in CDW, maintains a connection with CDP Base and all compaction jobs for the default Database Catalog run on CDP Base. However, subsequent Database Catalogs that are created do not maintain a connection to CDP Base and compaction runs entirely in CDW. Also in CDW, the worker process that performs the compaction work runs in HiveServer2, which equates to a Hive Virtual Warehouse. However, compaction performed by the worker process in Hive Virtual Warehouses consists of queries instead of MapReduce tasks.

Cloudera Data Warehouse Private Cloud Compaction Architecture

This diagram illustrates how the components that perform compaction interact on Cloudera Data Warehouse (CDW) Private Cloud. In CDW Private Cloud, all compaction tasks for the default Database Catalog are performed on CDP Base.



Considerations for using compaction on Cloudera Data Warehouse Private Cloud

The first Hive Virtual Warehouse you create in Cloudera Data Warehouse (CDW) Private Cloud for a Database Catalog (not including the default Database Catalog) is automatically set as the compactor and performs all compaction work for subsequent Virtual Warehouses (Hive or Impala) created under that Database Catalog.

Consequently, you must take into account the query workload for compaction when you create the first Hive Virtual Warehouse. You must make sure that the warehouse has adequate resources to handle the compaction workload in addition to any other workloads you might run in that warehouse.

Important:

- In the case of the default Database Catalog, all compaction takes place on CDP Base so you do not need to consider compaction queries for the Virtual Warehouses that use the default Database Catalog.
- Impala Virtual Warehouses cannot be designated as the compactor Virtual Warehouse for a Database Catalog. Compaction tasks can only be assigned to a Hive Virtual Warehouse.

Change compactor configuration for Hive Virtual Warehouses on Cloudera Data Warehouse Private Cloud

To enhance performance, the compactor is a set of background processes that compact delta files, which are created as a by-product of data modifications. When it runs, it incurs additional load on the Hive Virtual Warehouse assigned as the compactor in Cloudera Data Warehouse (CDW) Private Cloud. You can change which Hive warehouse performs compaction to load-balance this workload as necessary.

About this task

In CDW Private Cloud, data compaction is performed on HiveServer2, which equates to the Hive Virtual Warehouse construct in the UI. This means that compaction is essentially query execution. Compaction runs an INSERT statement created from the output of a SELECT statement and runs in the Hive Virtual Warehouse assigned as the compactor, thereby re-writing the data. The Hive Virtual Warehouse, configured as the compactor, delivers the query capacity to perform this. Therefore, when you size the Hive Virtual Warehouse that performs compaction, you must take into consideration the extra workload to run the compaction queries. That extra workload needs to be considered in addition to your other query workloads on the Hive Virtual Warehouse that is configured as the compactor.

Important: All compaction tasks for the warehouses that use the default Database Catalog, which is the Database Catalog automatically created for you when you activate an environment for CDW, are performed on CDP Base and do not affect the performance of Virtual Warehouses that use the default Database Catalog. For all other Database Catalogs that you create, you must consider the compaction query workload for the Hive Virtual Warehouse that performs compaction tasks.

Before you begin

One of the Hive Virtual Warehouses must be configured as the compactor for the associated Database Catalog (excluding the default Database Catalog whose commpaction is performed on CDP Base). This Hive Virtual Warehouse compactor runs all of the compaction queries for all Virtual Warehouses that use one particular Database Catalog, including Impala Virtual Warehouses. However, Impala Virtual Warehouses cannot be configured as the compactor Virtual Warehouse for a Database Catalog. Compaction tasks must be assigned to a Hive Virtual Warehouse. The first Hive Virtual Warehouse you create against a Database Catalog is automatically set as the compactor. If you decide you do not want that particular warehouse to take on the compaction workload, you can set another Hive Virtual Warehouse to perform the compaction workload by following these steps:

Procedure

1. Log in to the CDP web interface and navigate to the Data Warehouse service.

2. On the Overview page, select the Hive Virtual Warehouse that you want to set as the compactor, and click the options menu  .
3. In the options menu, select Set Compactor.

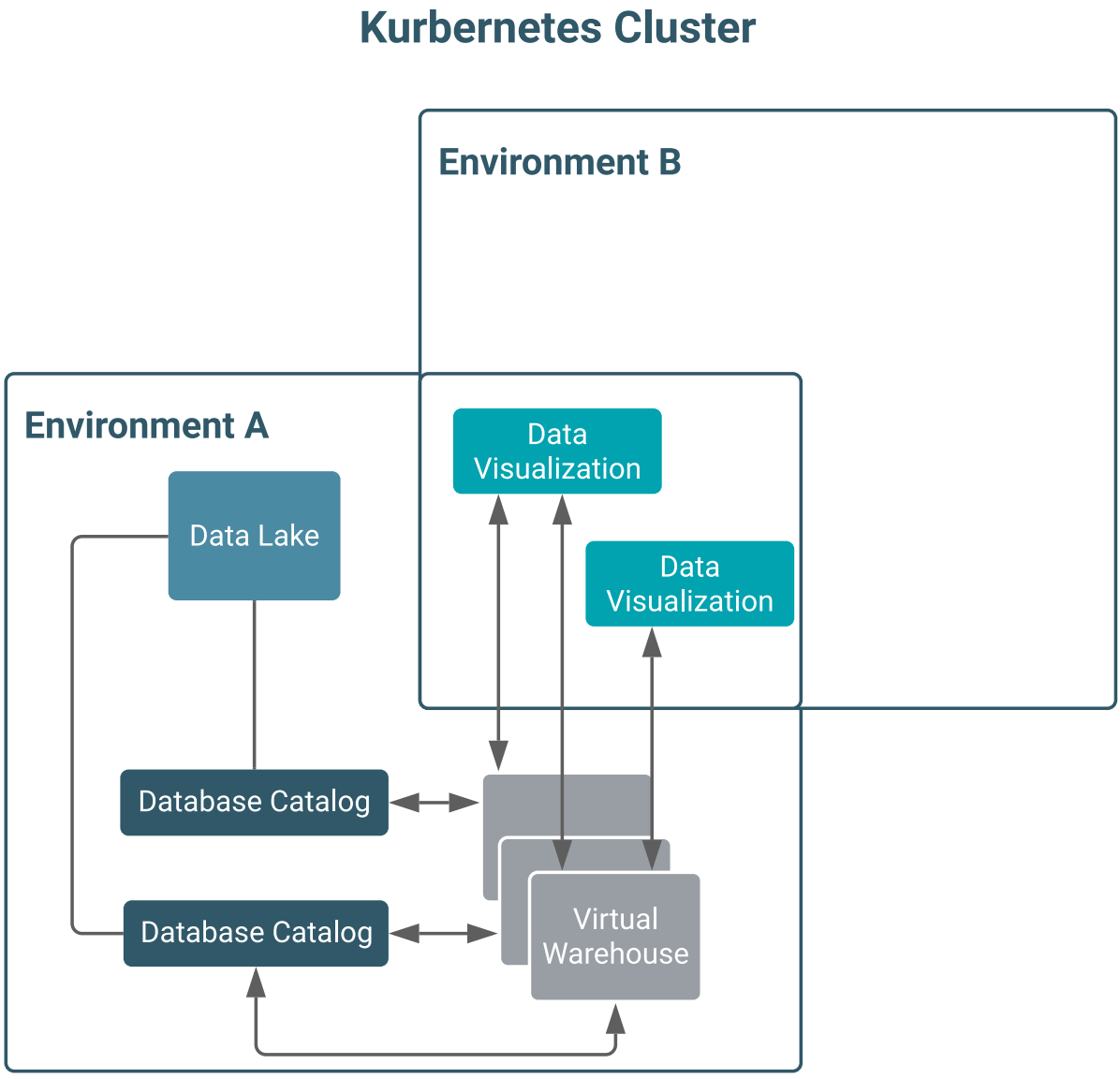
Related Information

CDW requirements for OpenShift

Data Visualization in Cloudera Data Warehouse

Cloudera Data Warehouse (CDW) integrates Data Visualization for building graphic representations of data, dashboards, and visual applications based on CDW data, or other data sources you connect to. You, and authorized users, can explore data across the entire CDP data lifecycle using graphics, such as pie charts and histograms. You arrange visuals on a dashboard for collaborative analysis.

You connect Data Visualization to a Virtual Warehouse as described in Starting Data Visualization integrated in CDW. Similar to using a BI client, you can configure and connect to Virtual Warehouses from different clusters. You configure the connection in a familiar way, providing an IP address or host name. Data Visualization is not tied to a particular Virtual Warehouse (VW). You can access data for your visualization from multiple Data Catalogs using multiple Hive or Impala Virtual Warehouses and multiple environments.



Having multiple Data Visualization instances attached to an environment, you can create dashboards for different groups. For example, Marketing and Sales can have their own private dashboards. When you delete a Virtual Warehouse, your visuals remain intact.

Related Information

Cloudera Data Visualization

Creating a visual

Working with datasets

Configuring Impala Virtual Warehouses to create Impala tables in Kudu in Cloudera Data Warehouse Private Cloud

Cloudera Data Warehouse allows you to create Impala tables in Kudu. You can configure an Impala Virtual Warehouse to connect to Kudu and create Impala tables in Kudu using Hue. Or, you can create tables on the fly by

specifying the Kudu master host in the TBLPROPERTIES statement while running the query from the Hue query editor.

About this task

Attention: This feature is in technical preview and not recommended for use in production environments.

Before you begin

Obtain the hostname of the Kudu master home by going to Cloudera Manager Clusters Kudu service Instances from the CDP Management Console.

Creating Imapla tables in Kudu on the fly

To create Impala tables in Kudu without updating a Virtual Warehouse's Impala coordinator configuration, you must specify the Kudu master host in the TBLPROPERTIES statement as follows while running the query from Hue:

```
TBLPROPERTIES ('kudu.master_addresses'='[***host.example.com***]')
```

Configuring the Virtual Warehouse to create Impala tables in Kudu

By reconfiguring an existing Impala Virtual Warehouse as follows, any tables you create will be created in Kudu.

Procedure

1. Log in to the Cloudera Data Warehouse service as a DWAdmin.
2. Go to an Impala Virtual Warehouse and click ⋮ Edit CONFIGURATIONS Impala coordinator and select flagfile from the drop-down list.
3. Click + and enter the following key and value:

Key	Value
kudu_master_hosts	[***HOSTNAME-OF-KUDU-MASTER***]

4. Click APPLY.
5. Restart the Virtual Warehouse.
6. Open Hue from the same Virtual Warehouse.
7. Enter the following lines in the query editor and click the run button:

```
# Create a new table
# Use the kudu.num_tablet_replicas if the Kudu cluster is too small
CREATE TABLE my_first_table
(
  id BIGINT,
  name STRING,
  PRIMARY KEY(id)
)
PARTITION BY HASH PARTITIONS 16
STORED AS KUDU
TBLPROPERTIES ('kudu.num_tablet_replicas' = '1');
# Insert into Kudu table
INSERT INTO my_first_table VALUES (99, "sarah");

# Verify if the data was inserted
```

```
SELECT * FROM my_first_table;
```

The above commands create an Impala table in Kudu and insert a sample record. The following is a screenshot showing the SQL commands and their output in Hue: