

Using BI tools with Cloudera Data Warehouse

Date published: 2024-01-01

Date modified: 2024-12-05



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

Built-in BI clients and drivers.....	4
Connecting to Impala Virtual Warehouse from Impala shell client.....	4
Using the Impala shell client.....	5
Download Beeline tarball.....	6
Connect to Hive from Tableau.....	7
Connect a Virtual Warehouse and Microsoft Power BI Desktop.....	11
Download JDBC JAR.....	13
Upload additional JARs.....	14

Built-in BI clients and drivers in Cloudera Data Warehouse

You can connect BI tools and SQL clients such as Beeline, impala-shell, impyla, Tableau, and so on to Cloudera Data Warehouse and use them to explore and query data in the Data Lakehouse.

Cloudera Data Warehouse provides built-in downloadables such as Hive JDBC driver for connecting JDBC-compliant tools to the Virtual Warehouses, Beeline CLI, Impala JDBC and ODBC drivers for connecting to Impala Virtual Warehouses, and a JDBC driver for using with Unified Analytics. You can download these from the Resources and Downloads tile present on the **Overview** page.

Connecting to Impala Virtual Warehouse from Impala shell client

You need to provide commands to your client users for installing and launching the Impala shell to connect to your Impala Virtual Warehouse. Client users can then query your tables. You learn how to obtain the command for installing the Impala shell on a client and other information to provide to clients.


About this task

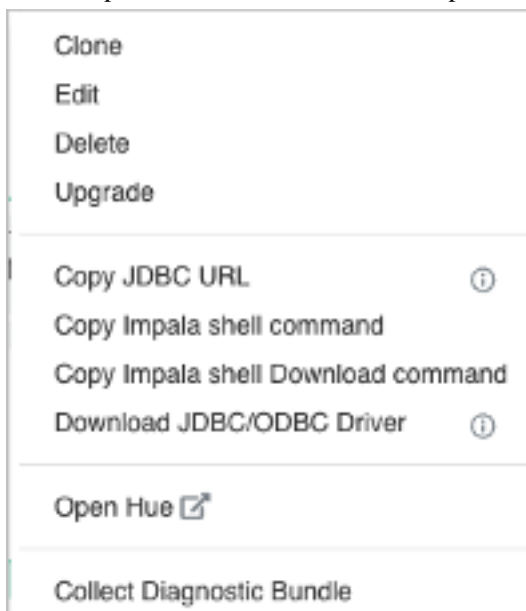
To query an Impala Virtual Warehouse from the Impala Shell, clients need to connect to HiveServer (HS2), which is available in the same cluster as Impala. The JDBC URL for the HS2 endpoint uses the strict HS2 protocol option to access the Impala Virtual Warehouse from the Impala Shell.

Before you begin

- Obtain the DWUser role.
- You must run a Virtual Warehouse version DWX 1.1.2-b2008 or later.

Procedure


1. In the Impala Virtual Warehouse, click Options  , and select Copy Impala shell Download command.



The command for installing the Impala shell compatible with your Impala Virtual Warehouse is copied to the clipboard. The command looks something like this:

```
pip install impala-shell==4.1.0a1
```

Using this command, your clients update impyla to the version compatible with Cloudera Data Warehouse, as listed in the Cloudera Data Warehouse Release Notes in section, “[Version mapping](#)”.

2. Click Options  again, and select Copy Impala shell command.

This action copies the command that launches the Impala shell and includes the connection string your client needs to connect to your Virtual Warehouse. For example:

```
impala-shell --protocol='hs2-http' --ssl -i 'coordinator-vw-impala.dw-dwx-rzs556.xcu2-8y8x.dev.clldr.work:443' -u client_max -l
```

3. Provide the commands you copied to your client user.
4. Provide the instructions in the next topic Using the Impala Shell client to your client user.

Using the Impala shell client

You obtain the commands from the Impala Virtual Warehouse owner for installing and launching the Impala shell to connect to the Impala Virtual Warehouse. You learn how to install and launch the Impala shell that connects to the Impala Virtual Warehouse UI. You then query tables in the Impala Virtual Warehouse, assuming you have authentication credentials. Windows clients are not supported.

Before you begin

- You must have the latest stable version of Python 3.0 or above.
- You must obtain the pip installer compatible with the Python version.

Procedure

1. In a terminal window, run the command provided by the Impala Virtual Warehouse owner to update impyla to the version compatible with the Impala Virtual Warehouse.

For example:

```
pip install impyla==0.18a2
```

Installing/updating to impyla 0.18a2, for example, is required before you install the Impala shell in the next step.

2. Run the Impala shell Download command provided by the Impala Virtual Warehouse owner to install the Impala shell.

The command looks something like this:

```
pip install impala-shell==4.4.0.a1
```

The Impala shell is installed.

3. Run the Help command to confirm a successful installation:

```
impala-shell --help
```

4. Run the provided Impala shell command to launch the Impala shell.

The command looks something like this:

```
impala-shell --protocol='hs2-http' --ssl -i 'coordinator-vw-impala.dw-dwx-rzs556.xcu2-8y8x.dev.cldr.work:443' -u client_max -l
```

The Impala shell is launched and you connect to the Impala Virtual Warehouse.

5. Run a SQL command to confirm that you are connected to the Impala Virtual Warehouse instance.

```
SHOW DATABASES;
```

The query returns a list of databases in the Impala Virtual Warehouse.

Downloading the Beeline CLI tarball

Download the Beeline CLI tarball from Cloudera Data Warehouse to your local system and use the Beeline client to connect to a Hive Virtual Warehouse and run queries. The archive file contains all the dependent JARs and libraries that are required to run the Beeline script.


Before you begin


From the Cloudera Management Console user profile, note the Workload User Name and Workload Password.



Attention: Ensure that Java is installed on the node on which you want to download and use the Beeline CLI, and you have set the JAVA_HOME environment variable correctly while installing the JDK.

Procedure

1. Log in to the Cloudera web interface and navigate to the Cloudera Data Warehouse service.
2. In the **Overview** page of the Cloudera Data Warehouse, click See More in the Resources and Downloads tile.
3. Select Beeline CLI and click  to download the file.
4. Save the apache-hive-beeline-x.x.xxxx.tar.gz file in your local system and extract the tarball.

5. In the Data Warehouse service **Overview** page, for the Virtual Warehouse you want to connect to the client, click  and select Copy JDBC URL.
6. Paste the copied JDBC URL in a text file, to be used in later steps.

```
jdbc:hive2://<your-virtual-warehouse>.<your-environment>.<dwx.company.com>/default;transportMode=http;httpPath=cliservice;ssl=true;retries=3
```

7. Open a terminal window and go to the folder where the tarball is extracted to start Beeline.
bin/beeline

This starts an interactive Beeline shell where you can connect to Hive and run SQL queries.

8. Run the connect command to connect to Hive using the JDBC URL that you copied earlier.

```
beeline> !connect [***JDBC URL***]
```

```
Connecting to jdbc:hive2://<your-virtual-warehouse>.<your-environment>.<dwx.company.com>/default;transportMode=http;httpPath=cliservice;ssl=true;retries=3
```

9. Enter the Workload User Name and Workload Password when you are prompted for the user credentials.

```
Enter username for jdbc:hive2://<your-virtual-warehouse>.<your-environment>.<dwx.company.com>/default: [***WORKLOAD USERNAME***]
Enter password for jdbc:hive2://<your-virtual-warehouse>.<your-environment>.<dwx.company.com>/default: [***WORKLOAD PASSWORD***]
Connected to: Apache Hive (version 3.1.2000.7.0.2.2-24)
Driver: Hive JDBC (version 3.1.2000.7.0.2.2-24)
Transaction isolation: TRANSACTION_REPEATABLE_READ
```

10. To verify if you are connected to HiveServer2 on the Virtual Warehouse, run the following SQL command:
SHOW TABLES;

```
INFO : Compiling command(queryId=hive_20200214014428_182d2b63-a510-421f-8bbc-65a4ae24d1d6): show tables
INFO : Semantic Analysis Completed (retrial = false)
INFO : Completed compiling command(queryId=hive_20200214014428_182d2b63-a510-421f-8bbc-65a4ae24d1d6); Time taken: 0.054 seconds
INFO : Executing command(queryId=hive_20200214014428_182d2b63-a510-421f-8bbc-65a4ae24d1d6): show tables
INFO : Starting task [Stage-0:DDL] in serial mode
INFO : Completed executing command(queryId=hive_20200214014428_182d2b63-a510-421f-8bbc-65a4ae24d1d6); Time taken: 0.018 seconds
INFO : OK
-----
table_name
-----
-----
No rows selected (0.311 seconds)
```

Connecting to Hive Virtual Warehouses from Tableau

This topic describes how to connect to Tableau with Hive Virtual Warehouses on Cloudera Data Warehouse service.


About this task

Required role: DWUser

Before you begin

Before you can use Tableau with Hive Virtual Warehouses, you must have populated your Database Catalog with sample data when you create it. You must also create a Hive Virtual Warehouse, which is configured to connect to the Database Catalog that is populated with data.

Procedure

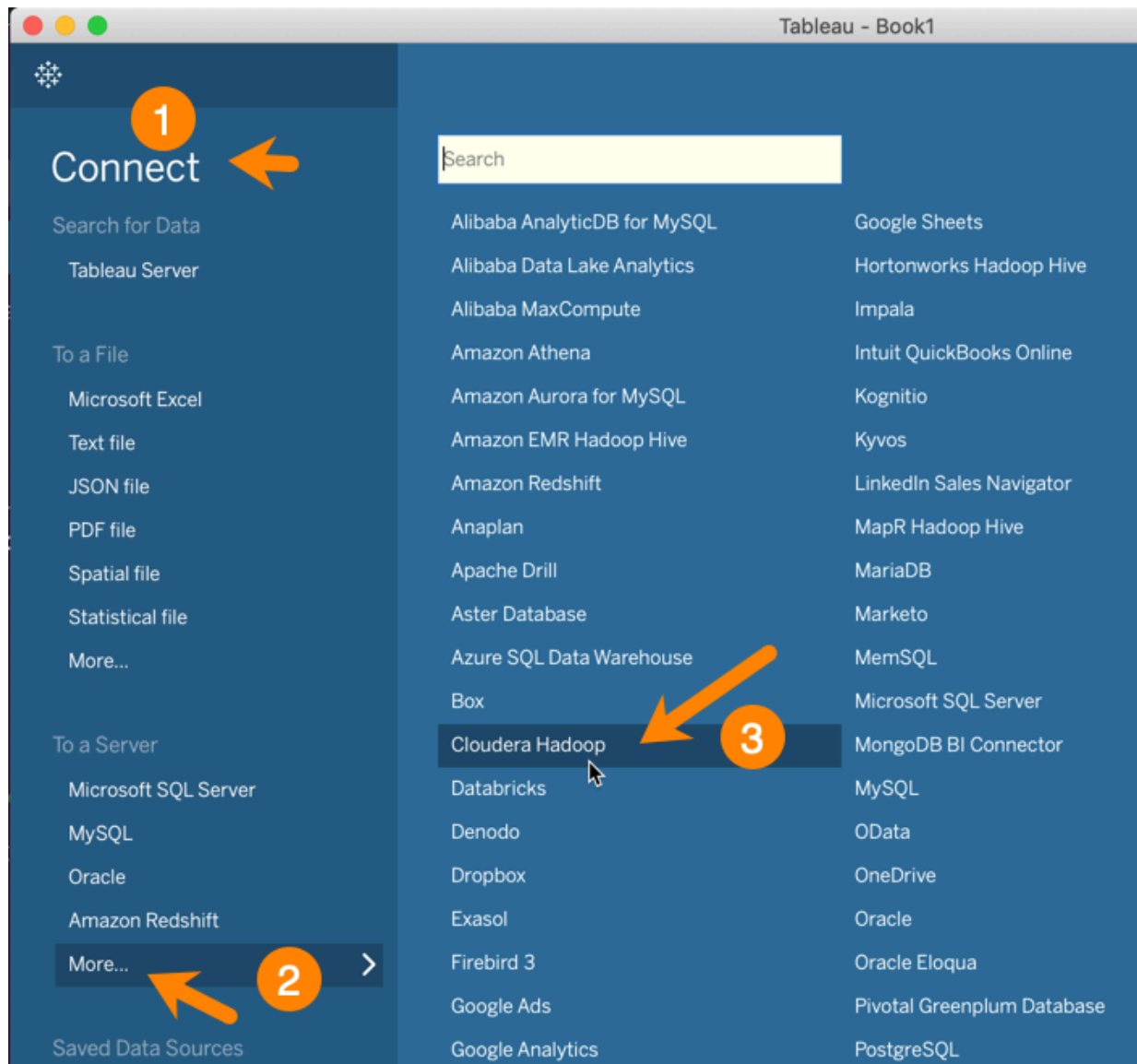
1. Download the latest version of the Hive ODBC driver from [Cloudera Downloads page](#).
2. Install the driver on the local host where you intend to use Tableau Desktop.
3. Log in to the Cloudera Data Warehouse service as DWUser.
4. Go to the **Virtual Warehouses** tab, locate the Hive Virtual Warehouse you want to connect to, and click  **Copy JDBC URL** .
This copies the JDBC URL to your system's clipboard.
5. Paste the copied JDBC URL into a text file. It should look similar to the following:

```
jdbc:hive2://<your-virtual-warehouse>.<your-environment>.<dwx.company.com>/default;transportMode=http;httpPath=cliservice;ssl=true;retries=3
```

6. From the text file where you just pasted the URL, copy the host name from the JDBC URL to your system's clipboard. For example, the host name in the URL is:

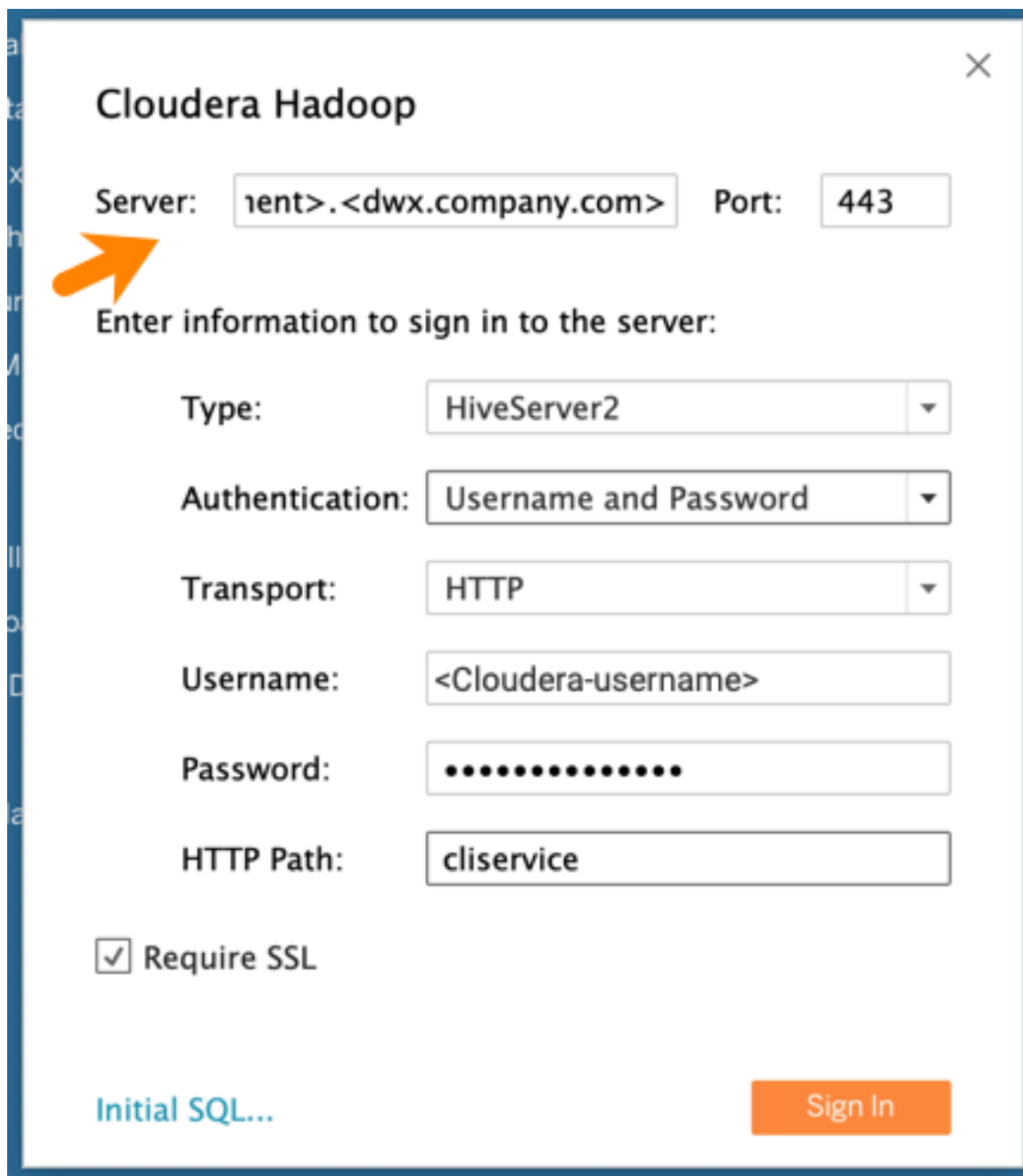
```
<your-virtual-warehouse>.<your-environment>.<dwx.company.com>
```


7. Start Tableau and navigate to ConnectMore...Cloudera Hadoop :



This launches the Cloudera Hadoop dialog box.

8. In the Tableau Cloudera Hadoop dialog box, paste the host name you copied to your clipboard in Step 7 into the Server field:



Cloudera Hadoop

Server: Port:

Enter information to sign in to the server:

Type:

Authentication:

Transport:

Username:

Password:

HTTP Path:

☒ Require SSL

[Initial SQL...](#) [Sign In](#)

9. Then in the **Tableau Cloudera Hadoop** dialog box, set the following other options:

- Port: 443
- Type: HiveServer2
- Authentication: Username and Password
- Transport: HTTP
- Username: Username you use to connect to the Cloudera Data Warehouse service.
- Password: Password you use to connect to the Cloudera Data Warehouse service.
- HTTP Path: cliservice
- Require SSL: Make sure this is checked.

10. Click Sign In.

Related Information

[Cloudera Hadoop connection option described in the Tableau documentation](#)

Connect a Virtual Warehouse and Microsoft Power BI Desktop

You learn how to connect to a Hive or Impala Virtual Warehouse from Microsoft Power BI. You can then use Power BI to visualize the data in the Virtual Warehouse on Cloudera.

About this task

Although you configure Power BI using a Microsoft UI labeled Hive LLAP, you can use this procedure to connect to a Hive Virtual Warehouse or to an Impala Virtual Warehouse. After making the connection, you can use Power BI to create reports based on data in the Virtual Warehouse.




Warning: We have not tested or certified connecting a Virtual Warehouse and Microsoft Power BI Desktop.

Before you begin

- You have access to a Cloudera Environment and a Hive LLAP or Impala Virtual Warehouse that contains tables of data.
- You know the Cloudera workload user name and password [you set in User Management](#) in the Cloudera Management Console. You need to use your workload user name and its associated password to log into the Virtual Warehouse.

Procedure

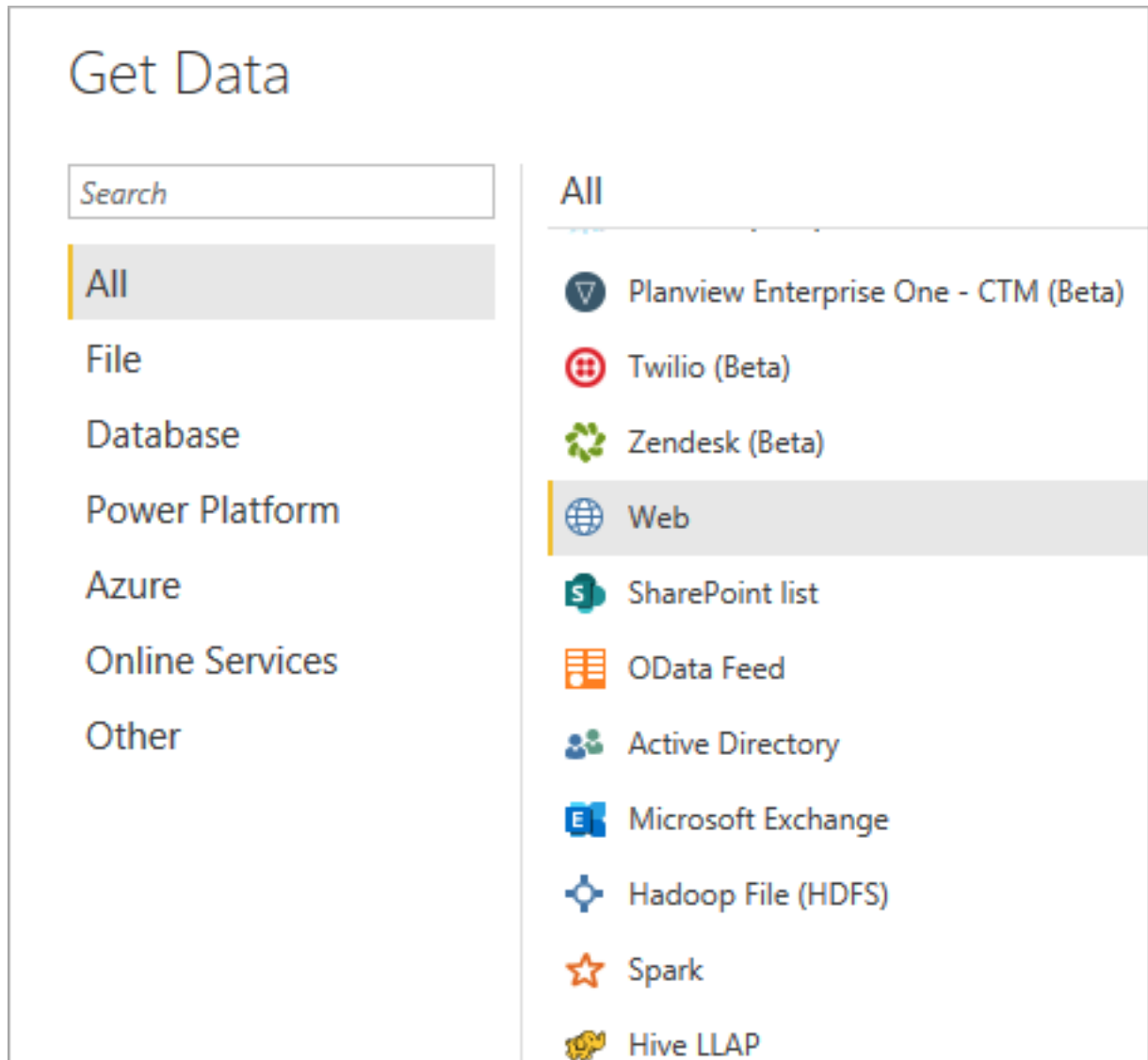
1. Log in to the Cloudera Data Warehouse service as DWUser.
2. Go to the **Virtual Warehouses** tab, locate the Hive Virtual Warehouse you want to connect to, and click  Copy JDBC URL .
This copies the JDBC URL to your system's clipboard.
3. Paste the copied JDBC URL into a text file. It should look similar to the following:

```
jdbc:hive2://<your-virtual-warehouse>.<your-environment>.<dwx.company.com>/default;transportMode=http;httpPath=cliservice;ssl=true;retries=3
```

4. From the text file where you just pasted the URL, copy the host name from the JDBC URL to your clipboard. For example, in the URL shown in the step above, the host name is:

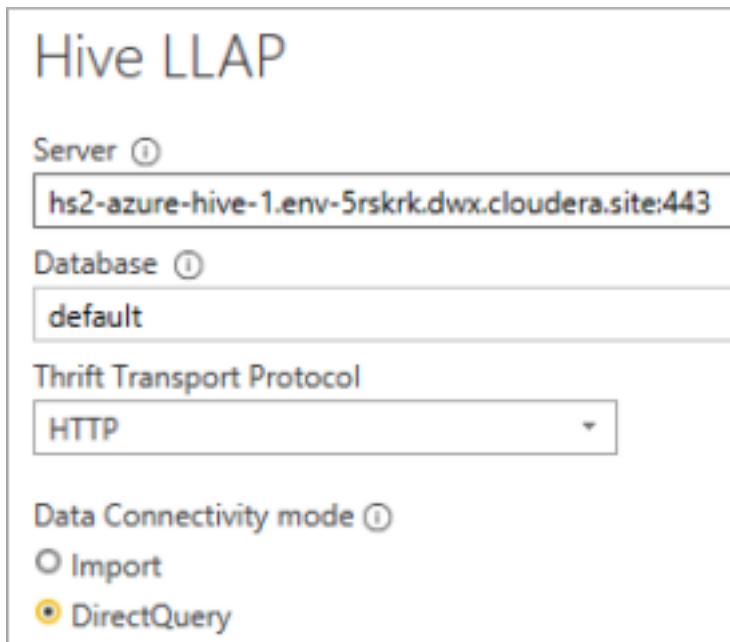
```
<your-virtual-warehouse>.<your-environment>.<dwx.company.com>
```

5. Open Power BI Desktop, select **Get Data More** .



6. Select Hive LLAP.

7. In the Hive LLAP dialog, configure Power BI to connect to the Hive or Impala Virtual Warehouse in Cloudera Public Cloud as follows:
 - In Server, paste the contents of your clipboard from step 4, add a colon and 443.
 - In Database, enter the database name, for example default.
 - In Thrift Transport Protocol, select HTTP.
 - In Data Connectivity mode, select DirectQuery.



Hive LLAP

Server ⓘ
hs2-azure-hive-1.env-5rskrk.dwx.cloudera.site:443

Database ⓘ
default

Thrift Transport Protocol
HTTP

Data Connectivity mode ⓘ
☐ Import
☒ DirectQuery

Click OK.

8. Enter your Cloudera workload user name and password.
9. Click Connect.

The PowerBI Navigator appears.

10. Select the tables you want to use in a report, and click Load.

In Fields, the table appears and is now available as a data source for your Power BI report.

Downloading a JDBC driver from Cloudera Data Warehouse

To use third-party BI tools, your client users need a JDBC JAR to connect your BI tool and the service. You learn how to download the JDBC JAR to give to your client, and general instructions about how to use the JDBC JAR.

Before you begin


Before you can use your BI tool with the Cloudera Data Warehouse service:

- You created a Database Catalog.

You have the option to populate your Database Catalog with sample data when you create it.
- You created a Virtual Warehouse and configured it to connect to the Database Catalog.

Of course, to query tables in the Virtual Warehouse from a client, you must have populated the Virtual Warehouse with some data.

Procedure

1. Log in to the Cloudera Data Warehouse service as DWUser.
2. In the **Overview** page of the Cloudera Data Warehouse service, click See More in the Resources and Downloads tile.
3. Click Hive JDBC Jar to download the Apache Hive JDBC JAR file.
4. Provide the JAR file you downloaded to your JDBC client.
On most clients, add the JAR file under the Libraries folder. Refer to your client documentation for information on the location to add the JAR file.
5. In the Cloudera Data Warehouse service **Overview** page, for the Virtual Warehouse you want to connect to the client, click  and select Copy JDBC URL.
A URL is copied to your system clipboard in the following format:

```
jdbc:hive2://<your_virtual_warehouse>.<your_environment>.<dwx.company.com>/default;transportMode=http;httpPath=cliservice;ssl=true;retries=3
```

6. Paste the URL into a text editor and configure your client BI tool to connect to the Virtual Warehouse using the following portion of the URL, represents the server name of the Virtual Warehouse:

```
<your_virtual_warehouse>.<your_environment>.<dwx.company.com>
```

7. In your JDBC client, set the following other options:
Authentication: Username and Password
Username: Username you use to connect to the Cloudera Data Warehouse service.
Password: Password you use to connect to the Cloudera Data Warehouse service.

Uploading additional JARs to Cloudera Data Warehouse

You add additional Java Archive (JAR) files to the Cloudera Data Warehouse Hive classpath that might be required to support dependency JARs, third-party Serde, or any Hive extensions.

About this task

- The JARs are added to the end of the Hive classpath and do not override the Hive JARs.
- Cloudera recommends that you do not use this procedure to add User-Defined Function (UDF) JARs. If you do, then you must restart HiveServer2 or reload the UDF. For more information about reloading functions, see the Hive Data Definition Language (DDL) manual.

Before you begin

You have the EnvironmentAdmin role permissions to upload the JAR to your object storage.

Procedure


1. Build the archive file.

The archive file can be either a .jar file or a tar.gz file. For a tar.gz archive file, only JARs present in the top level are considered.

For example, if the tar.gz file contains these files — test1.jar, test2.jar, and deps/test3.jar, only test1.jar and test2.jar are considered; deps/test3.jar is excluded.



Note: There is no defined priority to use a particular file. If there are multiple files with the same name or same class in multiple jars, any file can be in effect.

2. Upload the archive file to the Hive Virtual Warehouse on Cloudera Data Warehouse object storage, such as AWS S3 or Azure Data Lake Storage (ADLS).
3. Log in to the Cloudera Data Warehouse service and from the Overview page, locate the Hive Virtual Warehouse that uses the bucket or container where you placed the archive file, and click  and select Edit.
4. In the **Virtual Warehouse Details** page, click **Configurations Hiveserver2**.
5. From the Configuration files drop-down list, select env.
6. Search for CDW_HIVE_AUX_JARS_PATH and add the archive file to the environment variable.



Important: You can specify multiple locations in the CDW_HIVE_AUX_JARS_PATH environment variable separated by a colon (:). For example, s3a://examplebucket/path/to/file1.jar:s3a://examplebucket/path/to/file2.jar or abfs://storage@example/path/to/file1.jar:abfs://storage@example/path/to/file2.jar.




Important: Cloudera Data Warehouse only supports the default S3 bucket or ADLS Gen 2 container associated with the environment.

If you add a directory, the .jar or tar.gz files within the directory are copied and extracted. For a tar.gz file, only the JARs present in the top level are copied.

Consider the following JAR path - /common-jars/common-jars.tar.gz:/common-jars/single-jar.jar:/serde-specific-jar/serde.jar. In this example, common-jars.tar.gz is extracted and single-jar.jar and serde.jar files are copied.

7. Repeat the previous step and add the archive file or directory for Query coordinator and Query executor.

If the CDW_HIVE_AUX_JARS_PATH environment variable is not present, click  and add the following custom configuration:

```
CDW_HIVE_AUX_JARS_PATH=[ ***VALUE*** ]
```

8. If you require the JARs for Hive metastore (HMS), go to the corresponding Database Catalog and add the environment variable in **Configurations Metastore**.

Results

On applying the configuration changes, Hive Virtual Warehouse restarts and the archive files are available and added to the end of the Hive classpath.

Related Information

[Hive Data Definition Language manual](#)