

cloudera

Contents

1. About Cloudera	3
1.1. Cloudera Director	3
1.2. Cloudera Manager	4
2. Objective	4
3. Architecture	Error! Bookmark not defined.
4. Getting Started	5
4.1 Accessing Cloudera Backend cluster details	5
4.2. Configure SOCKS Proxy	8
4.3. Configure Your Browser to Access Proxy	9
4.4. Accessing Cloudera Manager from Cloudera Director Web UI	12
4.5. Hue.....	18
4.6. Apache Spark (Run Spark App)	21
4.7. Viewing Jobs in UI	24
4.8. Hive	26
4.9. Impala	28
5. Power BI integration with Data Lake Store and Impala (Optional)	31
5.1 Integrating with Data Lake Store	31
5.2 Integrating with Impala	38
6. Reference	41
6.1 Configure SOCKS Proxy	41
6.2 Restart Cloudera Management Service	42
6.3 Error Messages While Running the Spark Job.....	45

1. About Cloudera

Cloudera is an open-source Apache Hadoop distribution, CDH (Cloudera Distribution Including Apache Hadoop) targets enterprise-class deployments of that technology.

Cloudera provides a scalable, flexible, integrated platform that makes it easy to manage rapidly increasing volumes and varieties of data in your enterprise. Cloudera products and solutions enable you to deploy and manage Apache Hadoop and related projects, manipulate and analyze your data, and keep that data secure and protected.

Cloudera develops a Hadoop platform that integrates the most popular Apache Hadoop open source software within one place. Hadoop is an ecosystem, and setting a cluster manually is a pain. Going through each node, deploying the configuration through the cluster, deploying your services, and restarting them on a wide cluster is a major drawback of distributed system and require lot of automation for administration. Cloudera developed a big data Hadoop distribution that handles installation and updates on a cluster in few clicks.

Cloudera also develop their own projects such as Impala or Kudu that improve hadoop integration and responsiveness in the industry.

1.1. Cloudera Director

Cloudera Director enables reliable self-service for using CDH and Cloudera Enterprise Data Hub in the cloud.

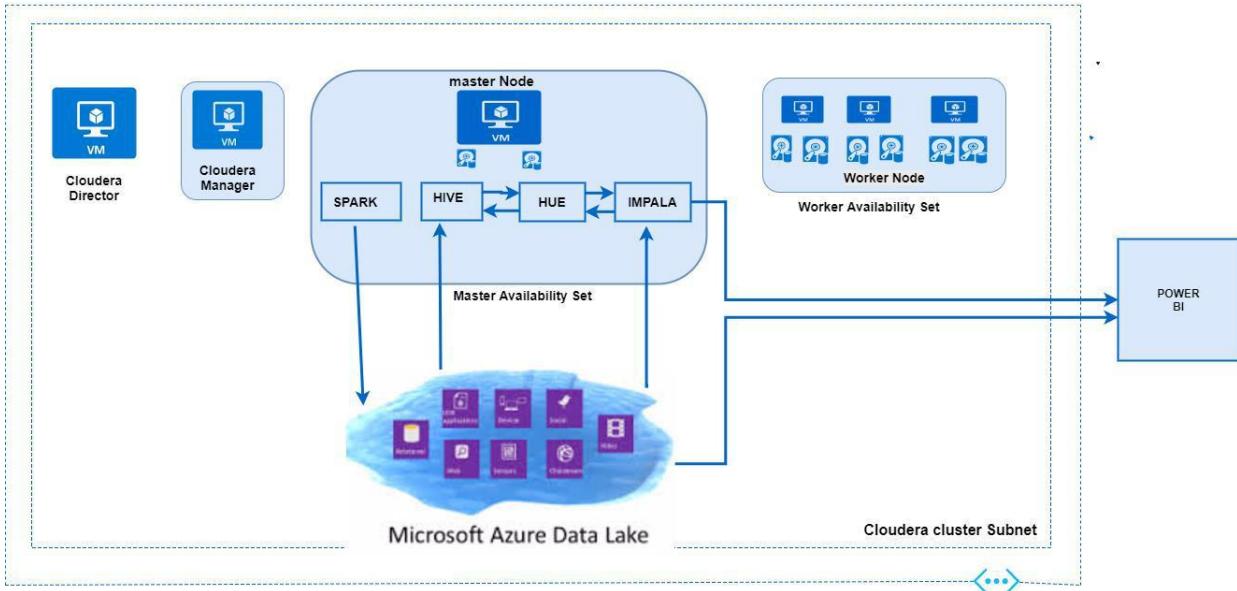
Cloudera Director provides a single-pane-of-glass administration experience for central IT to reduce costs and deliver agility, and for end-users to easily provision and scale clusters. Advanced users can interact with Cloudera Director programmatically through the REST API or the CLI to maximize time-to-value for an enterprise data hub in cloud environments.

Cloudera Director is designed for both long running and transient clusters. With long running clusters, you deploy one or more clusters that you can scale up or down to adjust to demand. With transient clusters, you can launch a cluster, schedule any jobs, and shut the cluster down after the jobs complete.

The Cloudera Director server is designed to run in a centralized setup, managing multiple Cloudera Manager instances and CDH clusters, with multiple users and user accounts. The server works well for launching and managing large numbers of clusters in a production environment.

1.2. Cloudera Manager

Cloudera Manager is a sophisticated application used to deploy, manage, monitor, and diagnose issues with your CDH deployments. Cloudera Manager provides the Admin Console, a web-based user interface that makes administration of your enterprise data simple and straightforward. It also includes the Cloudera Manager API, which you can use to obtain cluster health information and metrics, as well as configure Cloudera Manager.



2. Objective

NOTE: As this test drive provides access to the full Cloudera Director platform, deployment can sometimes take up to **45 minutes**. While you wait, please feel free to review helpful content in this manual and on Cloudera's [Azure Marketplace product page](#), or on the Cloudera [website](#). Please also consider watching the demo video showcased on the test drive launch page on the Azure Marketplace web site.

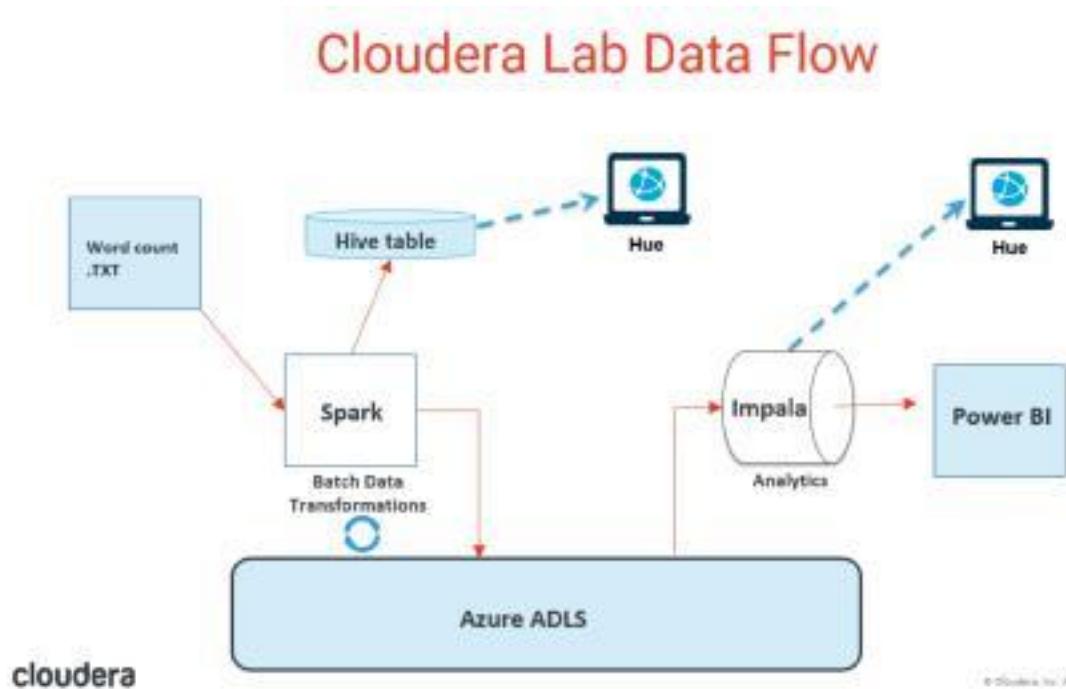
The test drive provisions Cloudera Director, the environment, Cloudera Manager, and a cluster consisting of 1 master node and 3 worker nodes. The test drive also integrates with Azure Data Lake Store.

The use case scenario for this test drive is to provide users with a test Azure Data Lake Store and:

1. Run the **WordCount** app with Hadoop/Spark on ADLS.
2. Create a Hive table on the output, and query Hive from Hue.

3. Create an Impala table on the ADLS output and query Impala from Hue or Power BI.

The following diagram shows how the data in this test case flows from a .TXT file via Hue to ADLS, processed by Spark.



3. Getting Started

Once you have signed up or signed in to the test drive from the Azure Marketplace, the test drive will deploy. Once it is finished deploying, you will receive all the necessary access information and credentials like URLs, usernames, and passwords, delivered via email.

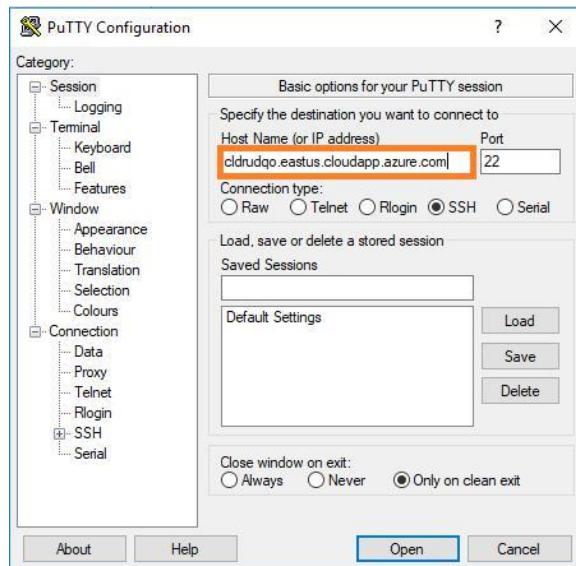
4.1 Accessing Cloudera Backend cluster details

Since the IP Address of the Master and Worker nodes change for each deployment, you must access Cloudera backend cluster details to get the Node Details.

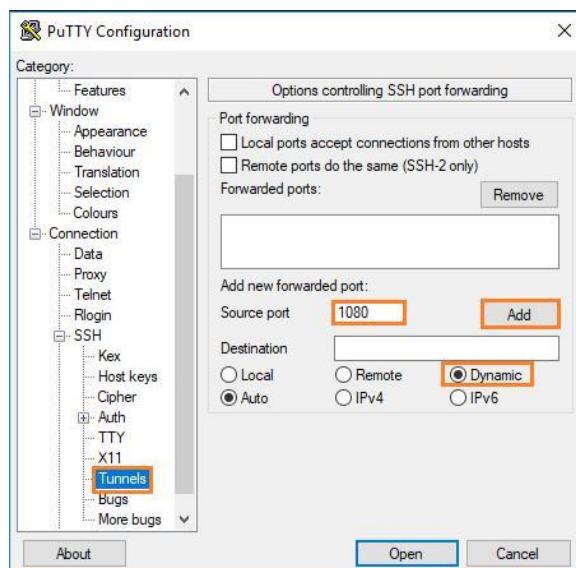
1. Log in to the Cloudera Director VM using the **Cloudera Director FQDN** address provided in the test drive access information, and an SSH tool like PuTTY (or Terminal on Mac), which we'll refer to in this walkthrough. ([Download PuTTY here](#))

E.g. cldrhyic.eastus.cloudapp.azure.com

* Mac users may visit section **6.1** in the **Reference** section in this guide to learn how to setup a tunnel using Terminal.



2. In the left navigation panel of PuTTY, navigate to **Connection > SSH > Tunnels**. Enter the **source port** as '**1080**', select **Dynamic**, click on **Add**, and then click **Open** to connect to the VM:



- Once connected, login to the Cloudera Director VM using the **Director Username** and then the **Director Password** from the provided test drive access credentials.
(Note: Passwords are hidden when typed or pasted in Linux terminals)



A screenshot of a terminal window titled 'cloudera@cldrudqo:~'. The window shows a login prompt: 'login as: cloudera', 'Using keyboard-interactive authentication.', and 'Password:'. The password field is completely blacked out.

- All the Cloudera Backend cluster details are present in NodeDetails file. **Copy the NodeDetails into a text file or Word document for reference**, these details will be used later.

To open the NodeDetails file use the following command.

```
cat NodeDetails
```

The NodeDetails file contains Node and URI details used by the Cloudera test drive environment. These are gathered using a script which pulls required data using the API calls.

```
[cloudera@cldrppyt ~]$ cat NodeDetails
Cloudera Director Node private IPAddress: 10.3.0.4
Cloudera Manager Node private IPAddress: 10.3.0.5
Cloudera Master Node private IPAddress: 10.3.0.9
Cloudera Hue Web UI URL: http:// 10.3.0.9:8888
Cloudera Hue Web UI Username/Password: admin/admin
Your Datalake Directory for the testdrive: demotdppyt
Your Datalake Endpoint for the testdrive: adl://cddatalakeppyt.azuredatalakestore.net
Your Output Data files on Datalake for the testdrive: adl://cddatalakeppyt.azuredatalakestore.net/demotdppyt/WordCount
```

4.2. Configure SOCKS Proxy

For security purposes, Cloudera recommends that you connect to your cluster using a [SOCKS proxy](#). A SOCKS proxy changes your browser to perform lookups directly from your Microsoft Azure network and allows you to connect to services using private IP addresses and internal fully qualified domain names (FQDNs).

This approach does the following:

- Sets up a single SSH tunnel to one of the hosts on the network (the Cloudera Director host) and create a SOCKS proxy on that host.

- Changes the browser configuration to do all lookups through that SOCKS proxy host.

Note: In the previous section (4.1), the SSH tunnel has already been setup on source port 1080.

- Run the following command in the terminal, which turns the Linux machine into Socks Proxy. (copy/paste may not work):

```
ssh -f -N -D 0.0.0.0:1080 localhost
```

When asked "Are you sure you want to continue connecting (yes/no)?", type "**yes**" and hit **Enter**. Then enter the **Director Password** mentioned in the provided access credentials for the test drive.

```
[cloudera@cldrppyt ~]$ ssh -f -N -D 0.0.0.0:1080 localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
RSA key fingerprint is 44:51:ae:e0:78:81:a6:93:d7:5c:bc:d7:79:9b:7f:ec.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
Password:
[cloudera@cldrppyt ~]$
```

Verify the Socks Proxy is running using command `ps -ef | grep ssh` on the console.

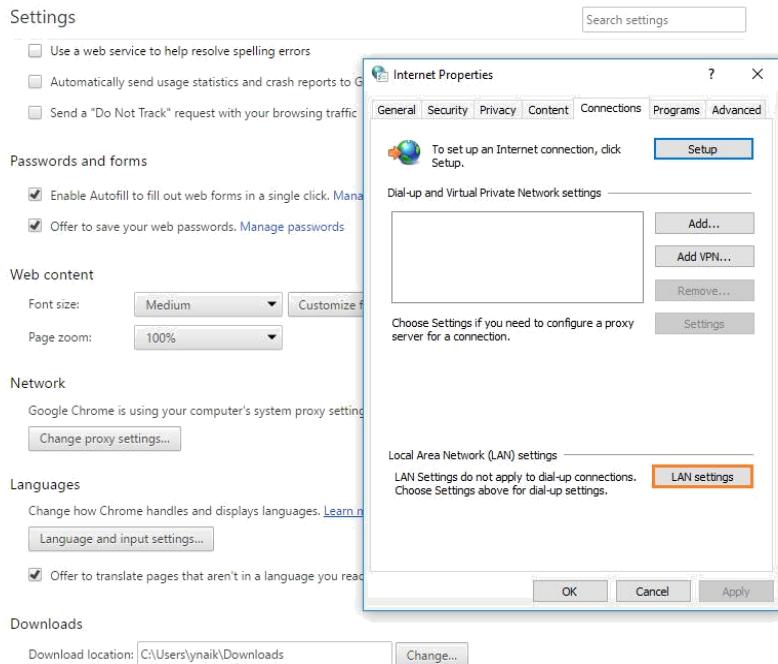
```
[cloudera@cldr2k45 ~]$ ps -ef | grep ssh
root      2137  1  0 00:26 ?        00:00:00 /usr/sbin/sshd
root      32775 2137  0 05:44 ?        00:00:00 sshd: cloudera [priv]
cloudera 32779 32775  0 05:44 ?        00:00:00 sshd: cloudera@pts/1
root      32905 2137  0 05:48 ?        00:00:00 sshd: cloudera [priv]
cloudera 32909  1  0 05:48 ?        00:00:00 ssh -f -N -D 0.0.0.0:1080 localhost
cloudera 32910 32905  0 05:48 ?        00:00:00 sshd: cloudera
root      32930 2137  0 05:52 ?        00:00:00 sshd: cloudera [priv]
cloudera 32932 32930  0 05:52 ?        00:00:00 sshd: cloudera@pts/0
cloudera 32933 32932  0 05:52 ?        00:00:00 /usr/libexec/openssh/sftp-server
root      33073 2137  0 05:58 ?        00:00:00 sshd: cloudera [priv]
cloudera 33103 33073  0 05:59 ?        00:00:00 sshd: cloudera@pts/2
cloudera 33129 33104  0 05:59 pts/2    00:00:00 grep ssh
```

4.3. Configure Your Browser to Access Proxy

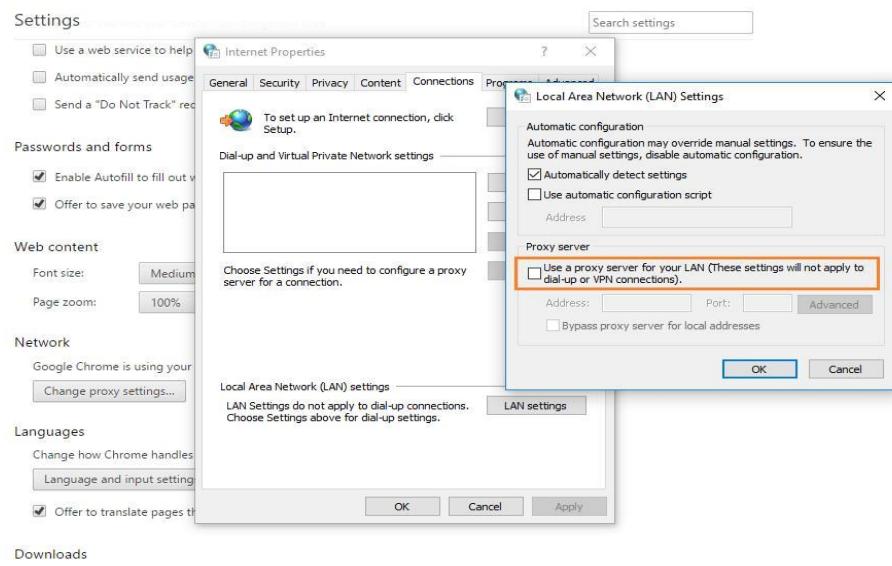
Follow the steps below to configure either a Chrome (1) or Edge (2) web browser to use a SOCKS proxy.

- Configuring for Chrome Web Browser:**

- Open Chrome settings and navigate to **Show advanced settings > Network > Change proxy settings**. Click on **LAN settings**.



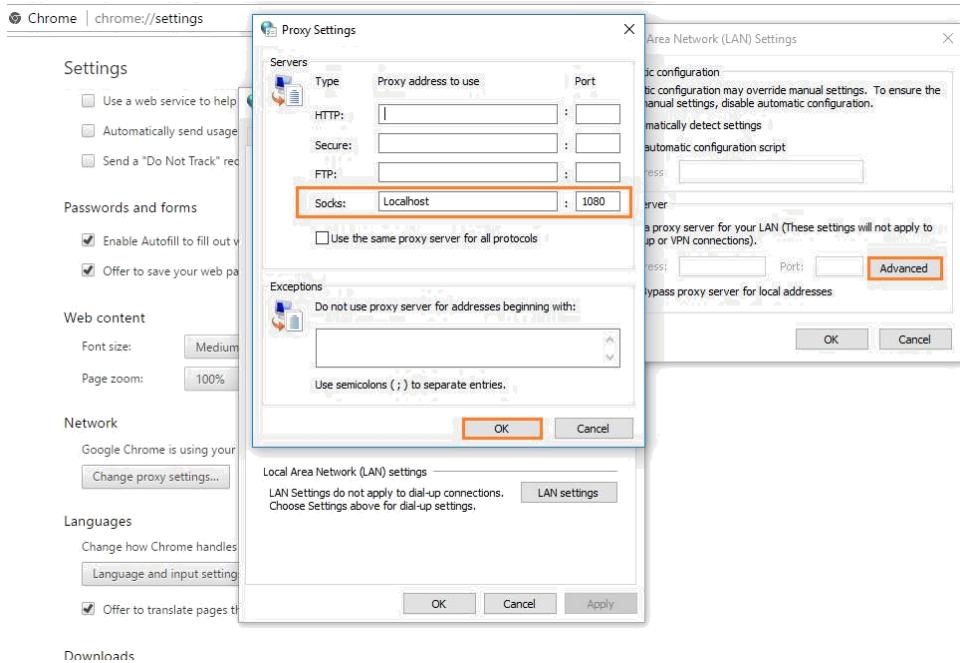
b. Check **Proxy Server** checkbox and click on **Advanced**.



c. Enter socks details:

Proxy address to use: **localhost**

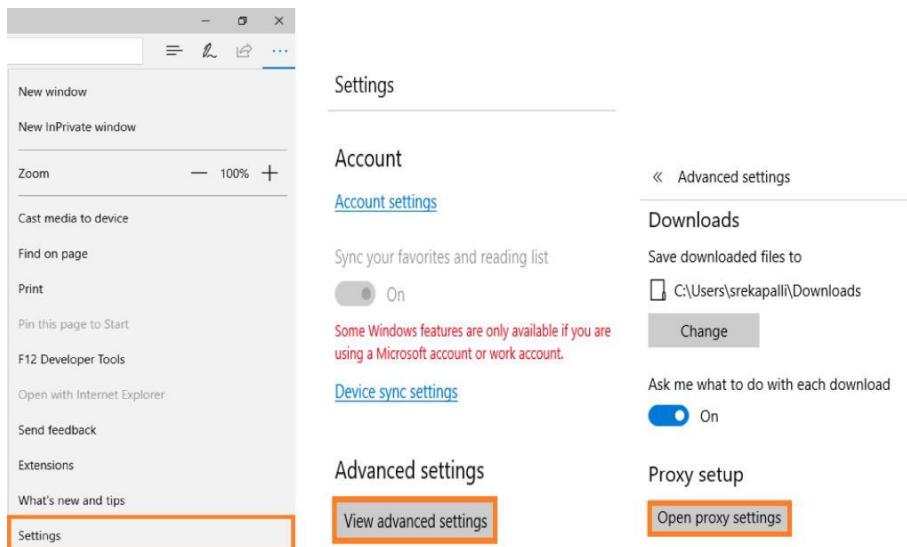
Port: **1080**



- d. Click on **OK** to save the changes.

2. Configuring for Edge Web Browser:

- a. Navigate to **Settings > View advanced settings > Open proxy settings.**



- b. Set "Use a proxy server" to **On**. Enter **Address** as <http://localhost> with **port** as **1080** and click on **Save**.

Proxy

Manual proxy setup

Use a proxy server for Ethernet or Wi-Fi connections. These settings don't apply to VPN connections.

Use a proxy server

On

Address

http://localhost

Port

1080

Use the proxy server except for addresses that start with the following entries. Use semicolons (;) to separate entries.

Don't use the proxy server for local (intranet) addresses

3. Configuring Chrome for Linux

```
/usr/bin/google-chrome \
--user-data-dir="$HOME/chrome-with-proxy" \
--proxy-server="socks5://localhost:1080"
```

4. Configuring Chrome for Mac OS X

```
"/Applications/Google Chrome.app/Contents/MacOS/Google Chrome" \
--user-data-dir="$HOME/chrome-with-proxy" \
--proxy-server="socks5://localhost:1080"
```

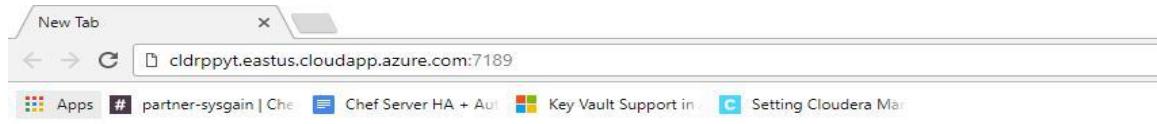
Note: Please visit section **6.1** in the **Reference** section later in this guide for additional details and help for any error messages you may encounter.

4.4. Accessing Cloudera Manager from Cloudera Director Web UI

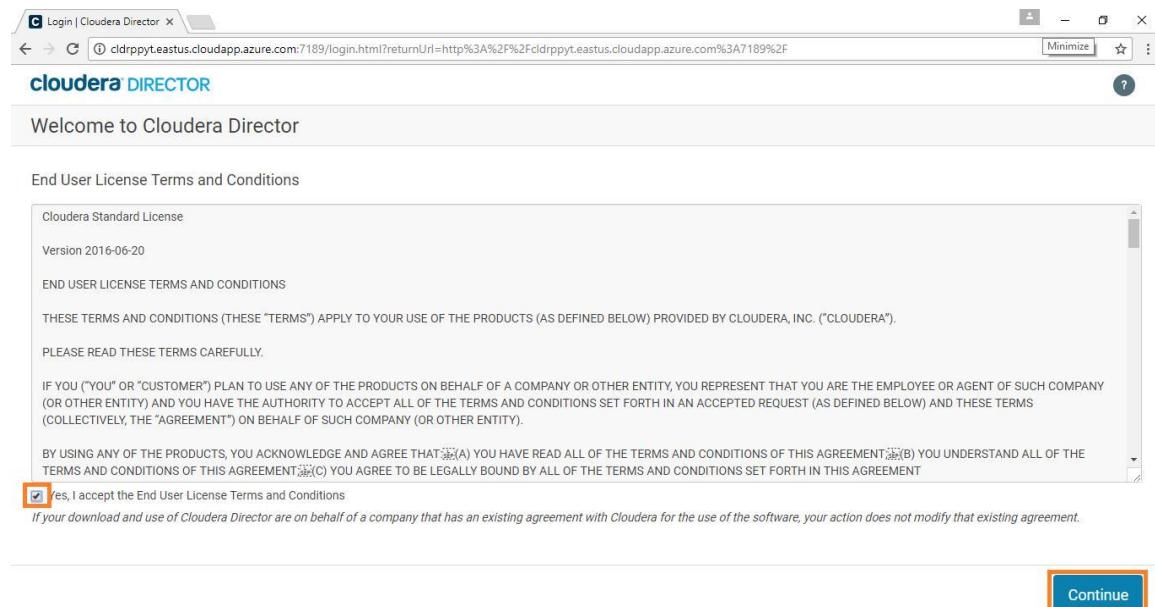
After deploying a cluster, you can manage it using Cloudera Manager:

1. Access the Cloudera Director Web UI using **Cloudera Director Access URL** provided in the Access Information.

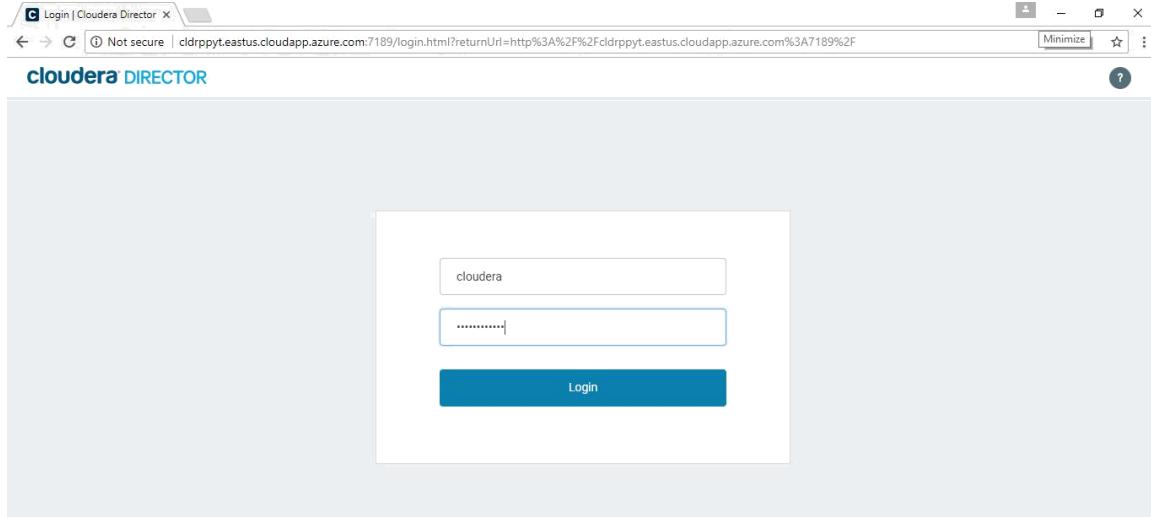
Eg: **cldrhyic.eastus.cloudapp.azure.com:7189**



2. **Accept the End User License Terms and Conditions** and click on **Continue**.



3. Login to the Cloudera Director web console using **CD-WEB UI Username** and **Password** from the Access Information.



4. The Cloudera Director console should open. Click on the **Cloudera Manager** link from the **Cloudera Director** Dashboard, as shown below.

5. Copy the Cloudera Manager **Host** address, along with the **port** number, and paste it in new browser tab.

Director_Azure_Deployment Environment

Director_Azure_Deployment Deployment

Deployment Details

Status	Ready	Host	10.3.0.5:7180
URL	Cloudera Manager	Instance	View Properties
Diagnostic Log Status		Not Collected	

Deployment Template

Instance Template	View Template	Configuration	Cloudera Manager Configurations
Image ID	cloudera-centos-6-latest	Provider Instance Type	STANDARD_DS14
License Type	Cloudera Enterprise Trial	Kerberos	Not Enabled
		Repository	https://archive.cloudera.com/cm5/redhat/6/x86_64/cm/5.11.1

Clusters

Cluster name	Status	Services	CDH version	Actions
Director_Azure_Deployment	Good health	Other	5.11.1-1.cdh5.11.1.p0.4	Modify Cluster

6. Login to the Cloudera Manager Console using **CM-WEB UI Username** and **CM-WEB UI Password** from the Access Information.

Log In - Cloudera Manager

Managed by Cloudera Director

cloudera MANAGER

Support Portal [Help](#)

admin

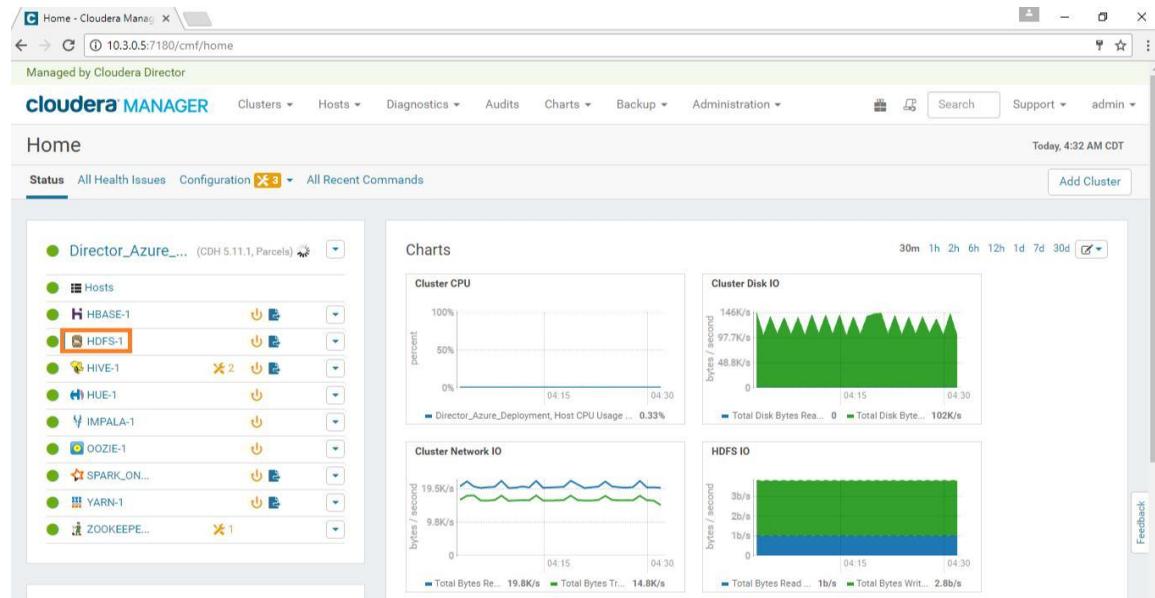
.....

Log In

Remember me

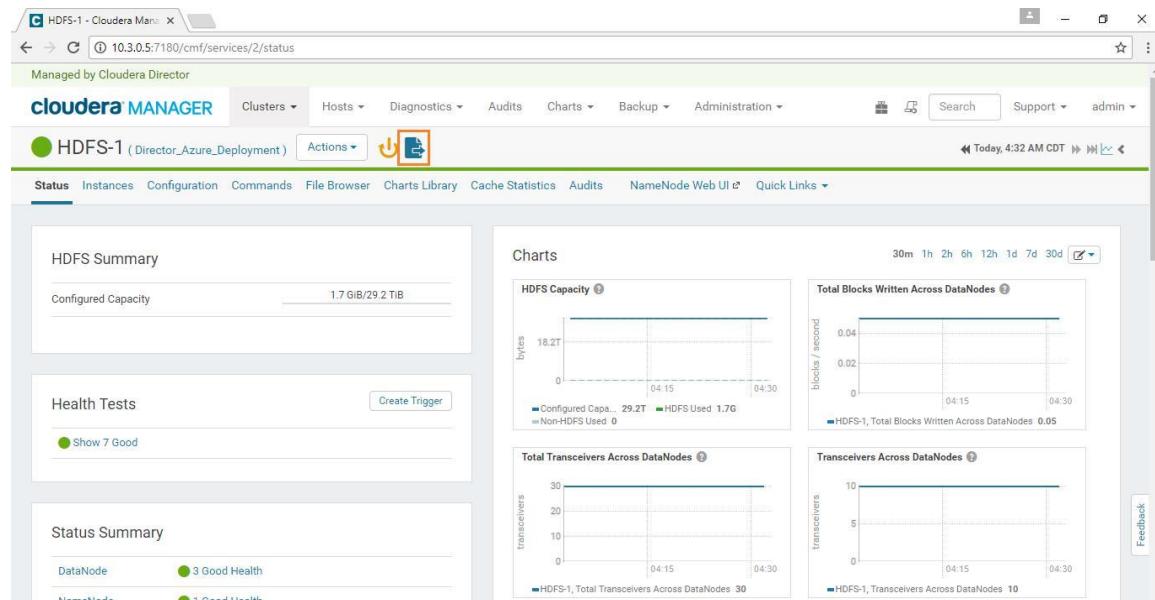
Note: The next step is to Restart Stale Services. We must do this to get the Azure Service Principle updated to the configuration file *site-core.xml*, which is required to integrate with Azure Data Lake Store.

7. In Cloudera Manager, click on the **HDFS-1** service to **Restart Stale Services**.



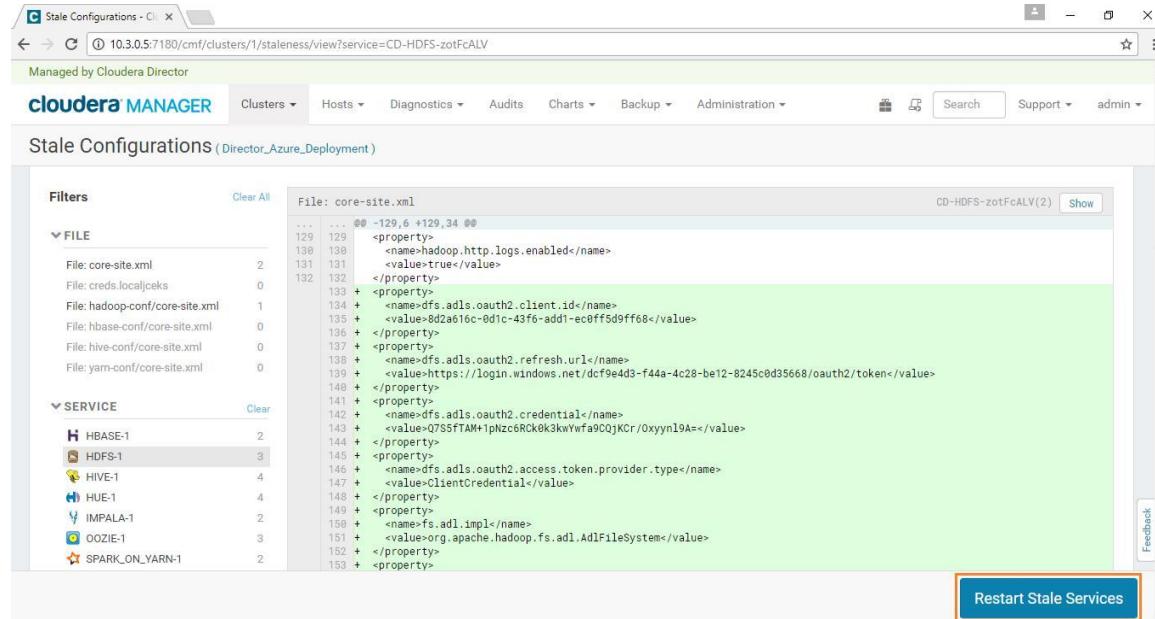
The screenshot shows the Cloudera Manager Home page. On the left, a list of services is displayed, with 'HDFS-1' highlighted and a red box drawn around it. The main area contains four charts: 'Cluster CPU', 'Cluster Disk IO', 'Cluster Network IO', and 'HDFS IO'. The 'Cluster CPU' chart shows Director_Azure_Deployment, Host CPU Usage at 0.33%. The 'Cluster Disk IO' chart shows Total Disk Bytes Read at 146K/s and Total Disk Bytes Written at 97.7K/s. The 'Cluster Network IO' chart shows Total Bytes Read at 19.8K/s and Total Bytes Written at 14.8K/s. The 'HDFS IO' chart shows Total Bytes Read at 3b/s and Total Bytes Written at 1b/s. The top right corner shows the date and time as 'Today, 4:32 AM CDT'.

8. Click on **Restart Stale Services** as shown in the below screenshot.



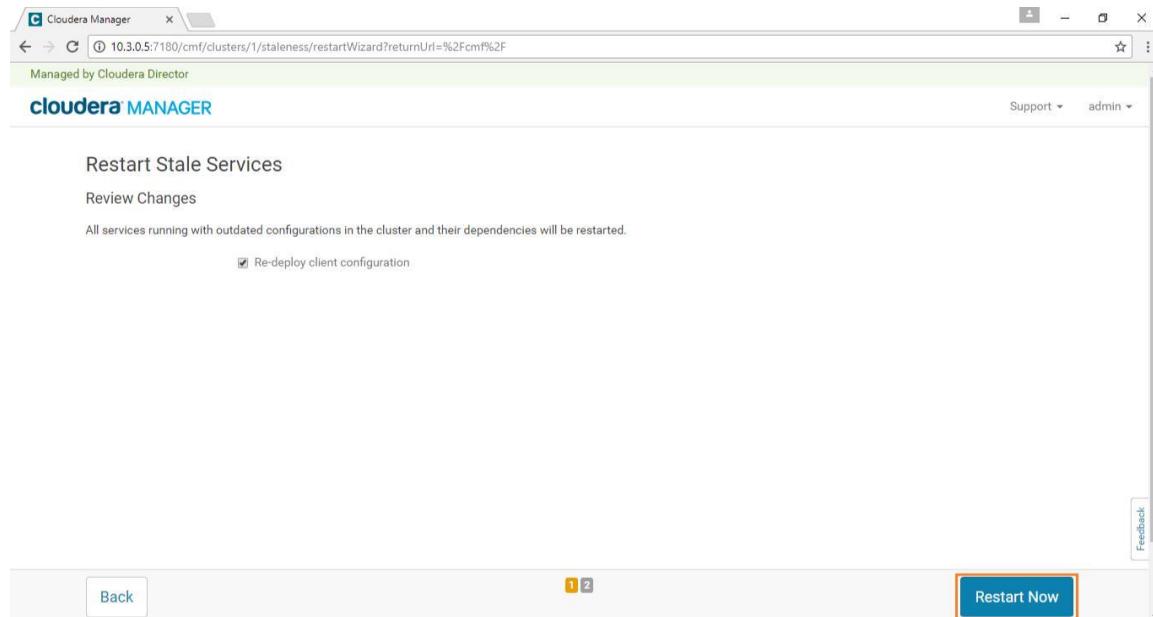
The screenshot shows the Cloudera Manager HDFS-1 service page. The 'Actions' dropdown menu is open, and the 'Restart Stale Services' option is highlighted with a red box. The page includes sections for 'HDFS Summary', 'Health Tests', and 'Status Summary', along with four charts: 'HDFS Capacity', 'Total Blocks Written Across DataNodes', 'Total Transceivers Across DataNodes', and 'Transceivers Across DataNodes'. The top right corner shows the date and time as 'Today, 4:32 AM CDT'.

9. Click on **Restart Stale Services** so the cluster can read the new configuration information.



The screenshot shows the Cloudera Manager interface for managing stale configurations. The top navigation bar includes 'Clusters', 'Hosts', 'Diagnostics', 'Audits', 'Charts', 'Backup', 'Administration', 'Search', 'Support', and 'admin'. The main content area is titled 'Stale Configurations (Director_Azure_Deployment)'. On the left, there are 'Filters' for 'FILE' and 'SERVICE'. Under 'FILE', 'core-site.xml' is selected, showing a diff view with changes in 'core-site.xml'. Under 'SERVICE', 'HDFS-1' is selected. At the bottom right, a blue button labeled 'Restart Stale Services' is highlighted with a red box.

10. Click on the **Restart Now** button.



The screenshot shows the 'Restart Stale Services' wizard. The top navigation bar includes 'Clusters', 'Hosts', 'Diagnostics', 'Audits', 'Charts', 'Backup', 'Administration', 'Search', 'Support', and 'admin'. The main content area is titled 'Restart Stale Services'. It shows a summary: 'Review Changes' and 'All services running with outdated configurations in the cluster and their dependencies will be restarted.' There is a checkbox for 'Re-deploy client configuration'. At the bottom right, a blue button labeled 'Restart Now' is highlighted with a red box.

11. Wait until all requested services are restarted. Once all the services are restarted, click on the **Finish** button.

Restart Stale Services

✓ Restart Awaiting Staleness Computation Command

Status: Finished Context: Director_Azure_Deployment Start Time: Jun 28, 4:35:42 AM Duration: 3.3m

All requested services successfully restarted.

Details Completed 2 of 2 step(s).

Step	Context	Start Time	Duration	Actions
➤ ✓ Execute global command Wait for configuration staleness computation Configuration staleness computation completed.	ε	Jun 28, 4:35:42 AM	50ms	
➤ ✓ Execute command Restart on cluster Director_Azure_Deployment All services successfully restarted.	ε Director_Azure_Deployment ε	Jun 28, 4:35:42 AM	3.3m	

Back 1 2 Finish Feedback

12. Now we have the **Cloudera Director** ready, with **Cloudera Manager** and **Cluster** (1 master and 3 workers).

Note: Please visit section **6.2** in the **Reference** section later in this guide for additional details and help for any error messages you may encounter.

4.5. Hue

Hue is a set of web applications that enable you to interact with a CDH cluster. Hue applications let you browse HDFS and manage a Hive metastore. They also let you run Hive and Cloudera Impala queries, HBase and Sqoop commands, Pig scripts, MapReduce jobs, and Oozie workflows.

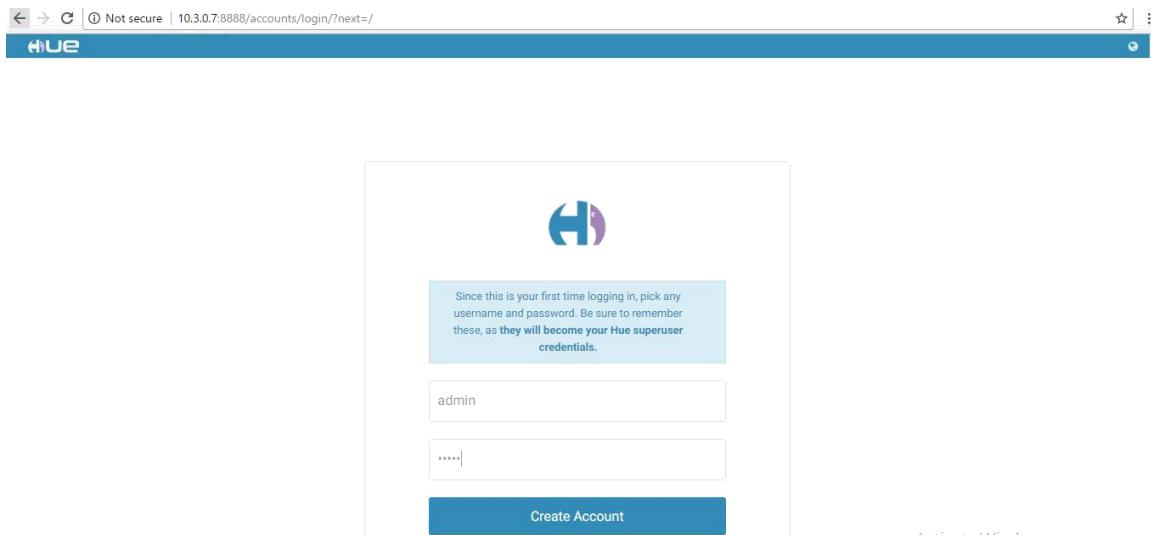
1. Copy the **Cloudera Hue Web URL** from the saved **NodeDetails** file and paste it in browser – which opens the Hue console.

10.3.0.7:8888

Apps Microsoft Azure Test

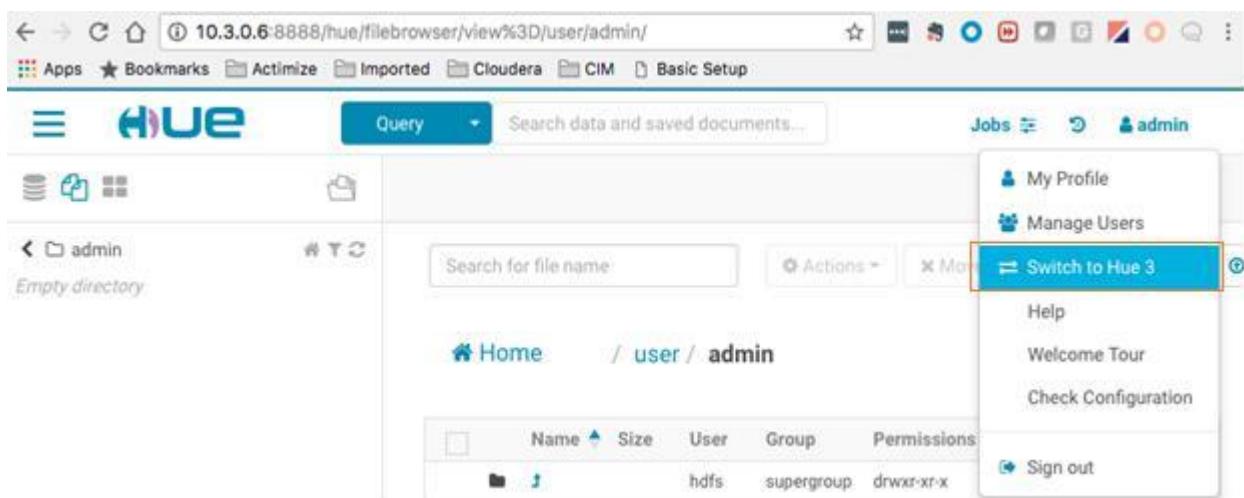
Gmail Images

2. Create a Hue Account by giving **Cloudera Hue Web UI Username/Password** from the **NodeDetails** file.



3. You will login into the Hue dashboard. On the right side of the page, click on the **HD**FS browser icon, as shown in the below screenshot.****

Note: CDH 5.12 has a new Hue UI. We recommend switching to Hue 3 from the admin tab (see screenshot below).



Quick Start Wizard - Hue™ 3.12 - The Hadoop UI

Step 1: **Check Configuration** Step 2: **Examples** Step 3: **Users** Step 4: **Go!**

Checking current configuration

Configuration files located in `/var/run/cloudera-scm-agent/process/85-hue-HUE_SERVER`

Potential misconfiguration detected. Fix and restart Hue.

PREFERRED_STORAGE_ENGINE We recommend MySQL InnoDB engine over MyISAM which does not support transactions.

OOZIE_EMAIL_SERVER Email notifications is disabled for Workflows and Jobs as SMTP server is localhost.

Activate Windows
Hue and the Hue logo are trademarks of Cloudera, Inc.

Back **Next**

4. Copy the data of **inputfile** from the below link. Give any name to the file (Eg: '**data**' or '**input**'), then save it in **.txt** format.

<https://aztdrepo.blob.core.windows.net/clouderadirector/inputfile.txt>

Once ready, click on **Upload** on the Hue file browser page (see below).

Note: Please ensure the inputfile is uploaded to the path **/user/admin** (see below):

File Browser

Search for file name Actions Move to trash

Upload New

Home **/ user / admin** History Trash

Name	Size	User	Group	Permissions	Date
hdfs		hdfs	supergroup	drwxr-xr-x	June 29, 2017 02:04 AM
admin		admin	admin	drwxr-xr-x	June 29, 2017 02:03 AM

5. Select the saved **.txt** file to upload.

The screenshots show the Hue File Browser interface. The top screenshot is a modal dialog titled 'Upload to /user/admin' with a 'Select files' button and a placeholder 'or drag and drop them here'. The bottom screenshot shows the main file browser view for the '/user/admin' directory. The file list includes:

Name	Size	User	Group	Permissions	Date
inputfile.txt	143 bytes	admin	admin	-rw-r--r--	June 29, 2017 02:10 AM
..		admin	admin	drwxr-xr-x	June 29, 2017 02:10 AM
hdfs		hdfs	supergroup	drwxr-xr-x	June 29, 2017 02:04 AM

6. The .txt file is now uploaded to Hue. The Spark application will use this data as input and provide the output to ADLS.

4.6. Apache Spark (Run Spark App)

Spark is the open standard for flexible in-memory data processing that enables batch, real-time, and advanced analytics on the Apache Hadoop platform.

To use it properly, it is also a good idea to install "dos2unix". dos2unix is a program that converts DOS to UNIX text file format, ensuring everything will run in a Linux environment.

1. Login to the **Master VM** by typing in the below command (**copy/paste may not work**):

```
ssh -i sshKeyForAzureVM cloudera@<Master Node private IPAddress>
```

Note: Provide **<Master Node private IPAddress>** from **NodeDetails** file.

```
[cloudera@cldrppy ~]$ ssh -i sshKeyForAzureVM cloudera@10.3.0.9
The authenticity of host '10.3.0.9 (10.3.0.9)' can't be established.
RSA key fingerprint is fb:d4:45:72:7b:2e:27:1f:74:c4:b1:0b:e5:79:26:6a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.3.0.9' (RSA) to the list of known hosts.
Last login: Fri Jun 30 02:07:10 2017 from cldrppy.cldrppy.local
[cloudera@cdmstr-9990c974 ~]$
```

2. **Download** the following script file using the below command.

The script contains the spark app (**WordCount**). The application counts the number of occurrences of each letter in words which have more characters than a given threshold.

```
wget
https://raw.githubusercontent.com/sysgain/clouderatd/master/scripts/ClouderaSparkSetup.sh
```

```
[cloudera@cdmstr-9990c974 ~]$ wget https://raw.githubusercontent.com/sysgain/clouderatd/master/scripts/ClouderaSparkSetup.sh
--2017-06-30 06:24:34-- https://raw.githubusercontent.com/sysgain/clouderatd/master/scripts/ClouderaSparkSetup.sh
Resolving raw.githubusercontent.com... 151.101.32.133
Connecting to raw.githubusercontent.com[151.101.32.133]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 815 [text/plain]
Saving to: "ClouderaSparkSetup.sh"

100%[=====] 815          --.-K/s  in 0s

2017-06-30 06:24:34 (139 MB/s) - "ClouderaSparkSetup.sh" saved [815/815]
[cloudera@cdmstr-9990c974 ~]$
```

3. To install **dos2unix**, run the following command:

```
sudo yum install -y dos2unix
```

```
[cloudera@cdmstr-9990c974 ~]$ sudo yum install -y dos2unix
Loaded plugins: fastestmirror, security
Setting up Install Process
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
--> Package dos2unix.x86_64 0:3.1-37.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved
```

4. To give permissions to **ClouderaSparkSetup.sh** file, run the following commands:

```
dos2unix /home/cloudera/ClouderaSparkSetup.sh
chmod 755 /home/cloudera/ClouderaSparkSetup.sh
```

```
[cloudera@cdmstr-9990c974 ~]$ dos2unix /home/cloudera/ClouderaSparkSetup.sh
dos2unix: converting file /home/cloudera/ClouderaSparkSetup.sh to UNIX format ...
[cloudera@cdmstr-9990c974 ~]$ chmod 755 /home/cloudera/ClouderaSparkSetup.sh
[cloudera@cdmstr-9990c974 ~]$
```

5. Run the following command to execute the **ClouderaSparkSetup.sh** script:

```
sh ClouderaSparkSetup.sh <Datalake Directory> <Master Node private
IPAddress> <inputfile.txt> <Datalake Endpoint for the testdrive>
```

Note: Replace the above values from **NodeDetails** and give the Name of the input file that you have just uploaded in Hue in the place of **<inputfile.txt>**.

Example: sh ClouderaSparkSetup.sh demotah6k 10.3.0.6 inputfile.txt
adl://cddatalakeah6k.azuredatastorage.net

```
[cloudera@cdmstr-9990c974 ~]$ sh ClouderaSparkSetup.sh demotdppyt 10.3.0.9 inputfile.txt adl://cddatalakeppyt.azuredatastorage.net
mkdir: `/user/cloudera': File exists
--2017-06-30 07:48:27-- https://aztdrepo.blob.core.windows.net/clouderadirector/wordcount.jar
Resolving aztdrepo.blob.core.windows.net... 52.238.56.168
Connecting to aztdrepo.blob.core.windows.net|52.238.56.168|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6371588 (6.1M) [application/octet-stream]
Saving to: "/home/cloudera/wordcount.jar"

2017-06-30 07:48:27 (1.00 MB/s) - "/home/cloudera/wordcount.jar" saved [6371588/6371588]
```

6. By executing the above script, the data has been stored to ADLS using Spark application.

Note: Please visit section **6.3** in the **Reference** section for additional details and help for any error messages you may encounter.

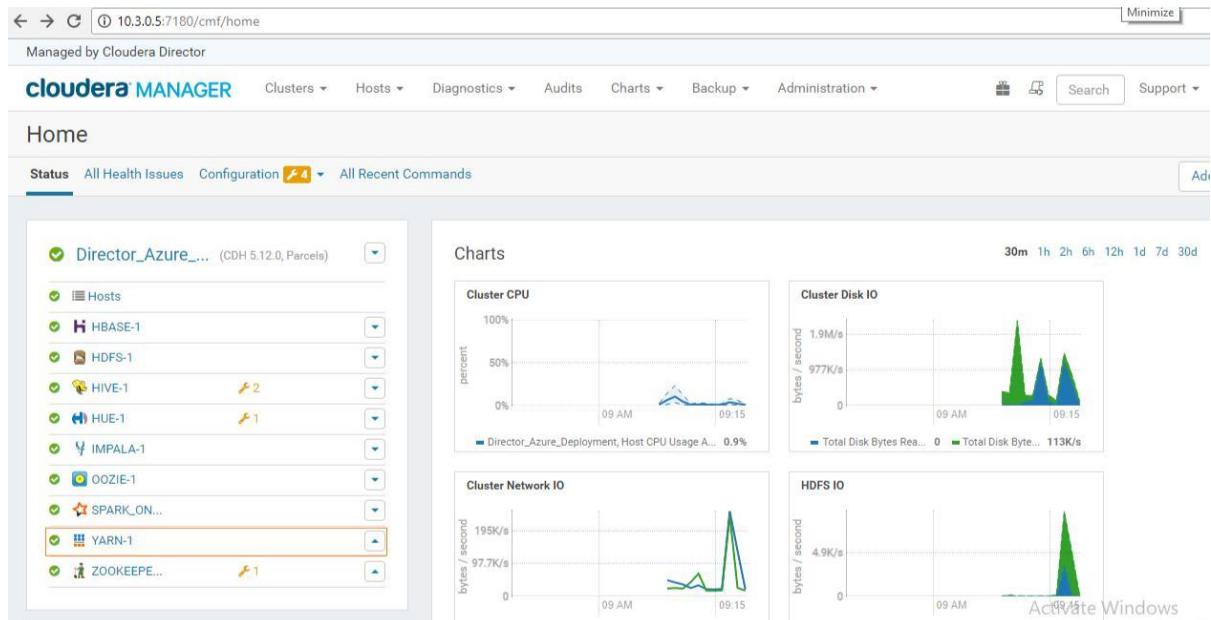
4.7. Viewing Jobs in UI

Next, navigate to the Yarn/Spark UI to see the WordCount Spark job.

1. Go to `http://<Manager Node private IP Address>:7180/cmf/home`

Example: <http://10.3.0.5:7180/cmf/home>

2. Click on **YARN-1**.



3. Click on **Applications** tab in the top navigation menu to view the available jobs.

Each job has Summary and Detail information. A job Summary includes the following attributes: **start & end timestamps**, **query name** (if the job is part of a Hive query), **queue**, **job type**, **job ID**, and **user**.

4. You can also see the available applications by navigating to the Spark UI:

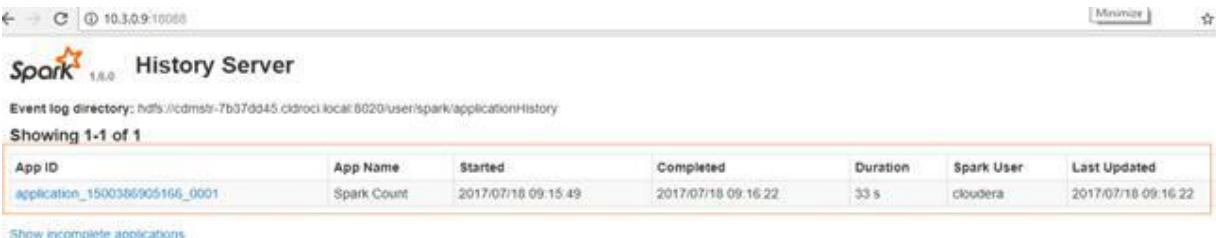
1. Go to <http://<Manager Node private IP Address>:7180/cmf/home>

Example: <http://10.3.0.5:7180/cmf/home>

2. Click on **SPARK_ON_YARN-1**. (May appear as '**SPARK_ON...**')

3. Navigate to the **History Server WEB UI** by going to `http://<Master Node private IP Address>:18088`

Example: <http://10.3.0.8:18088/>



The screenshot shows a browser window with the URL `http://10.3.0.9:18088` in the address bar. The page title is "Spark 1.6.0 History Server". Below the title, it says "Event log directory: /var/lib/cassandra/7ba7dd45.cassandra.local:8020/user/spark/applicationHistory". A table titled "Showing 1-1 of 1" displays one application entry:

App ID	App Name	Started	Completed	Duration	Spark User	Last Updated
application_150036905166_0001	Spark Count	2017/07/18 09:15:49	2017/07/18 09:16:22	33 s	cloudera	2017/07/18 09:16:22

Below the table, a link "Show incomplete applications" is visible.

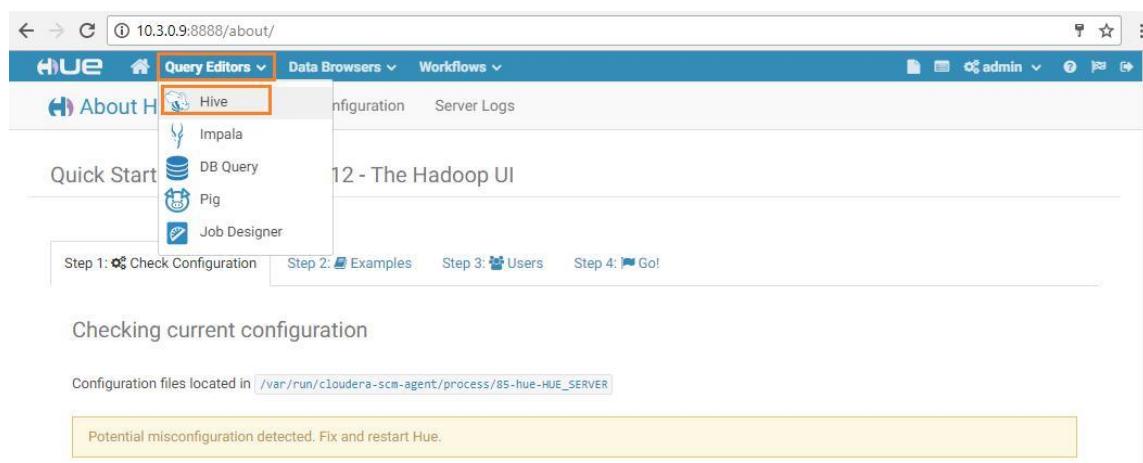
Note: Please visit section **6.3** in the **Reference** section for additional details and help for any error messages you may encounter.

4.8. Hive

Apache Hive is a data warehouse software project built on top of Apache Hadoop for providing data summarization, query, and analysis. Hive gives a SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop.

Now we will create a Hive table from the output of the Spark application stored on ADLS and run a Hive query from Hue.

1. Navigate to the **Query Editors** drop-down menu in the Hue WEB UI and click on **Hive**.

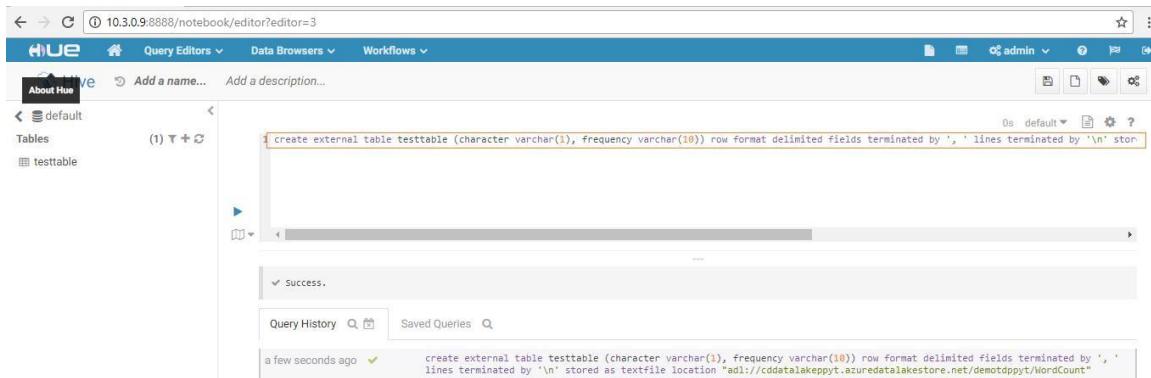


The screenshot shows a browser window with the URL `http://10.3.0.9:8888/about/` in the address bar. The page title is "HUE". The "Query Editors" dropdown menu is open, and the "Hive" option is selected, highlighted with a red box. The main content area shows "About Hue" with links to "Configuration" and "Server Logs". Below this, there is a "Quick Start" section with links to "Hive", "Impala", "DB Query", "Pig", and "Job Designer". At the bottom, there are four steps: "Step 1: Check Configuration", "Step 2: Examples", "Step 3: Users", and "Step 4: Go!". A message "Checking current configuration" is displayed. Below it, a message says "Configuration files located in /var/run/cloudera-scm-agent/process/85-hue-HUE_SERVER". A yellow box at the bottom contains the message "Potential misconfiguration detected. Fix and restart Hue.".

2. In the default database, execute the below query:

```
create external table <tablename> (character varchar(1), frequency  
varchar(10)) row format delimited fields terminated by ',' lines  
terminated by '\n' stored as textfile location "<Output Data files on  
Datalake for the testdrive>";
```

Note: Add any name for <tablename> and replace the <Output Data files on Datalake for the testdrive> placeholder with the corresponding data from the **NodeDetails** file.



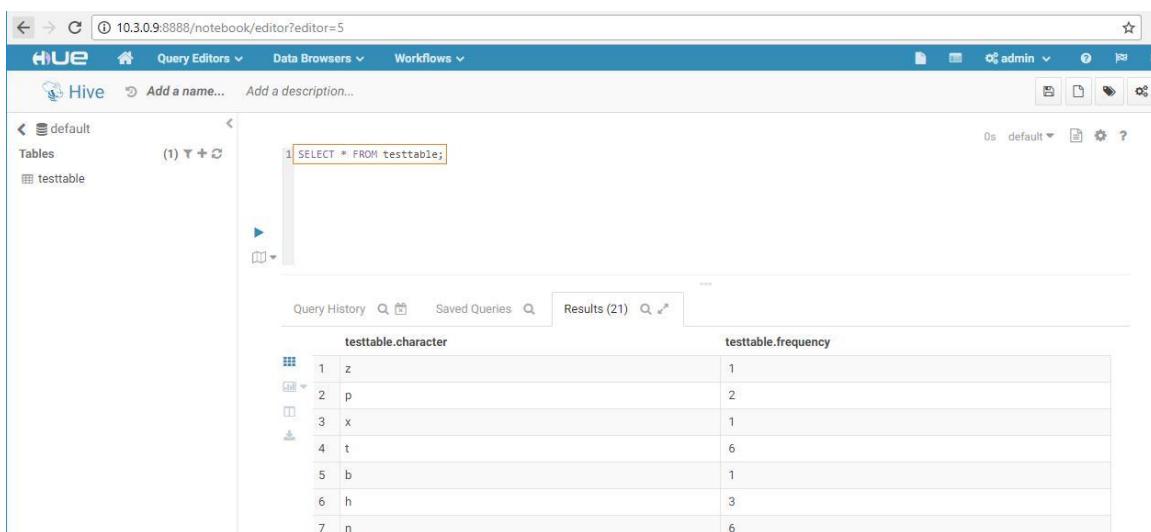
The screenshot shows the Hue Query Editor interface. The URL in the address bar is 10.3.0.9:8888/notebook/editor?editor=3. The top navigation bar includes 'HUE', 'Query Editors', 'Data Browsers', and 'Workflows'. The left sidebar shows a 'default' database with a 'Tables' section containing 'testtable'. The main query editor area contains the following SQL code:

```
create external table testtable (character varchar(1), frequency varchar(10)) row format delimited fields terminated by ',' lines terminated by '\n' stored as textfile location "adl://cdatalakeppyt.azuredatalakestore.net/demotoppyt/WordCount"
```

The status bar at the bottom indicates 'Success.' and shows the query was run 'a few seconds ago'. The 'Query History' tab is selected, showing the executed query.

3. View the table by giving the query:

```
Select * from <tablename>
```



The screenshot shows the Hue Query Editor interface. The URL in the address bar is 10.3.0.9:8888/notebook/editor?editor=5. The top navigation bar includes 'Hive', 'Query Editors', 'Data Browsers', and 'Workflows'. The left sidebar shows a 'default' database with a 'Tables' section containing 'testtable'. The main query editor area contains the following SQL code:

```
SELECT * FROM testtable;
```

The status bar at the bottom indicates 'Success.' and shows the query was run 'a few seconds ago'. The 'Results (21)' tab is selected, displaying the data from the 'testtable' table:

	testtable.character	testtable.frequency
1	z	1
2	p	2
3	x	1
4	t	6
5	b	1
6	h	3
7	n	6

4.9. Impala

Impala is an open source, massively parallel processing query engine on top of clustered systems like Apache Hadoop. It is an interactive SQL like query engine that runs on top of Hadoop Distributed File System (HDFS). It integrates with HIVE metastore to share the table information between both the components.

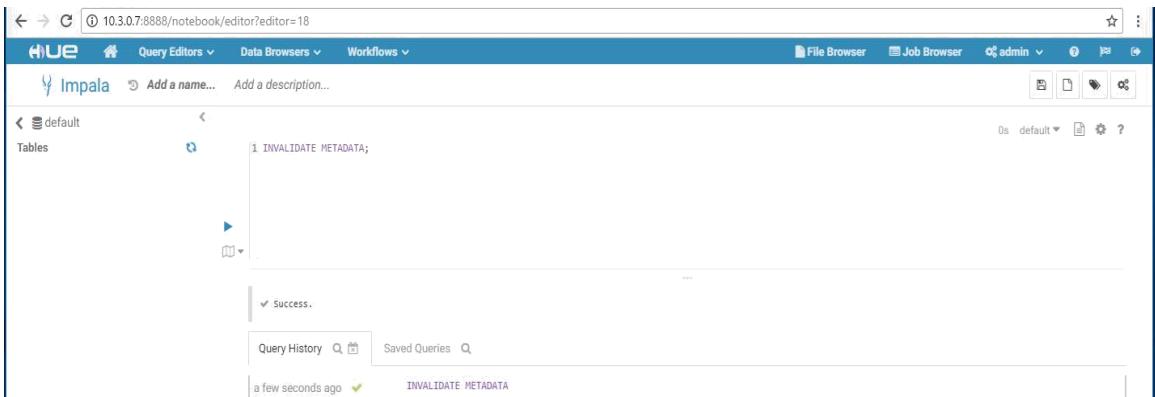
1. **Note:** Impala now integrates with ADLS from version CDH 5.12.

2. Navigate to the **Query Editor** drop-down menu and click on **Impala**.



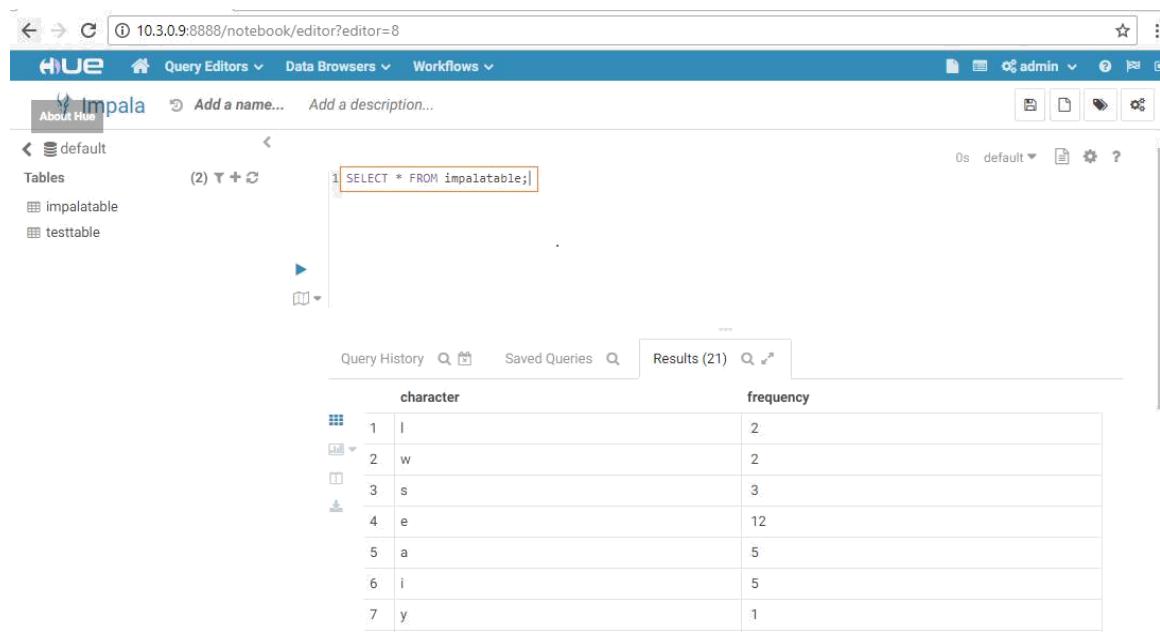
3. Execute the below query in the default database to sync the data from Hive to Impala:

```
INVALIDATE METADATA;
```



4. View the table by giving the query:

```
Select * from <tablename>
```



The screenshot shows the Hue interface for running Impala queries. The URL in the address bar is 10.3.0.9:8888/notebook/editor?editor=8. The top navigation bar includes 'Query Editors', 'Data Browsers', 'Workflows', and 'admin'. The left sidebar shows the 'default' database with tables 'impalatable' and 'testtable'. The main area contains a query editor with the following content:

```
1 SELECT * FROM impalatable;
```

Below the query editor is a table titled 'Results (21)'. The table has two columns: 'character' and 'frequency'. The data is as follows:

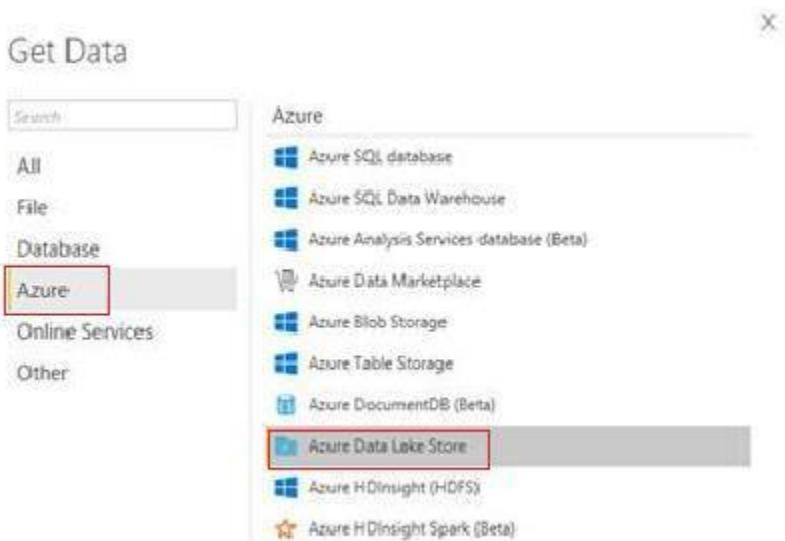
	character	frequency
1	l	2
2	w	2
3	s	3
4	e	12
5	a	5
6	i	5
7	y	1

5. You have now successfully run the Impala query using Hue!

4. Power BI integration with Data Lake Store and Impala (Optional)

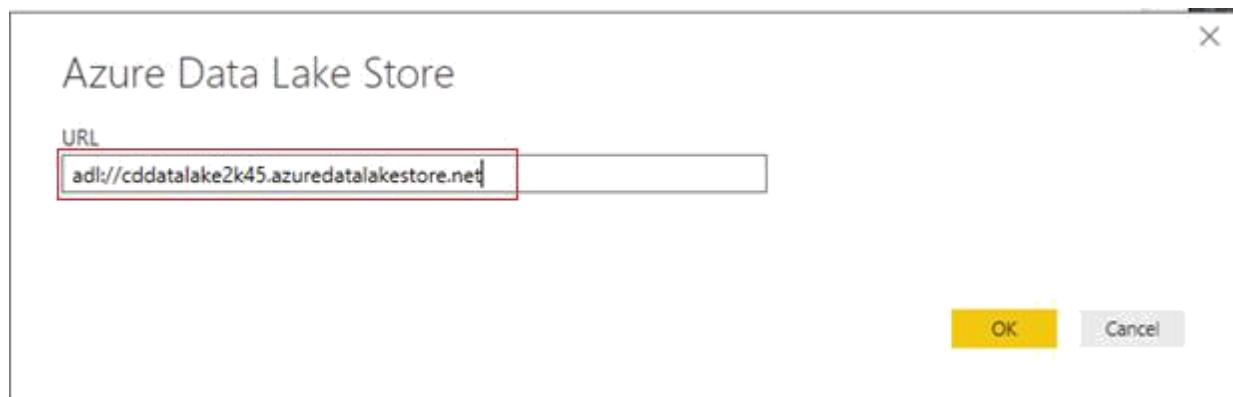
5.1 Integrating with Data Lake Store

1. Launch **Power BI Desktop** on your computer.
2. From the **Home** ribbon, click **Get Data**, and then click **More**. In the Get Data dialog box, click **Azure**, click **Azure Data Lake Store**, and then click **Connect**.



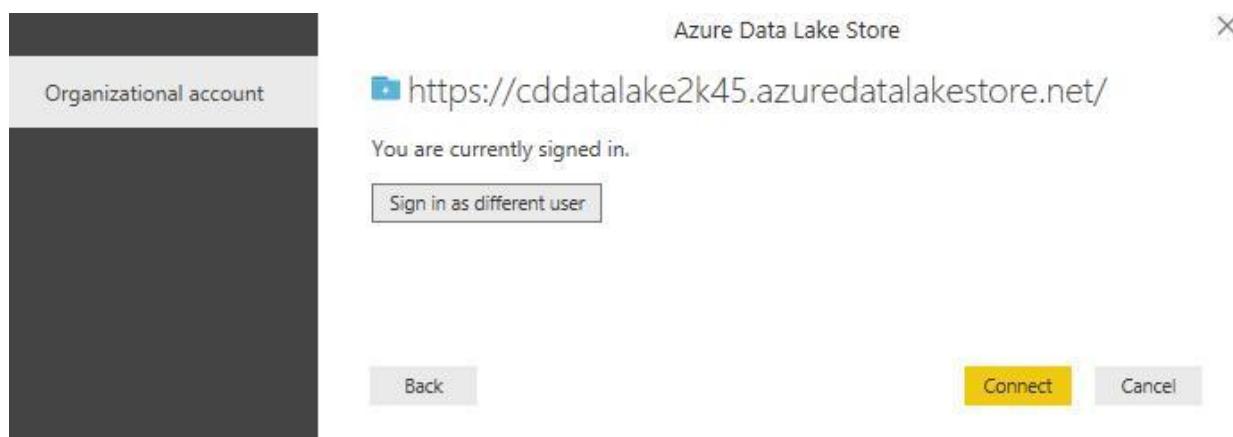
3. In the Microsoft Azure Data Lake Store dialog box, provide the **URL to your Data Lake Store account**, and then click **OK**.

Note: Get the **URL - Datalake Endpoint** from the NodeDetails file. (Refer to section **4.1**)

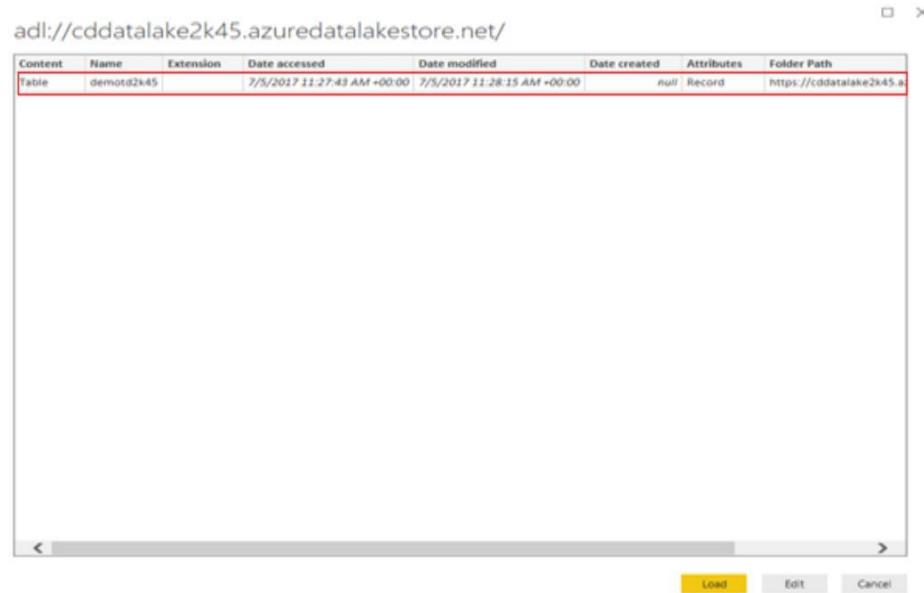


4. In the next dialog box, click **Sign in** to sign into Data Lake Store account. You will be redirected to your organization's sign in page. **Follow the prompts** to sign into the account.

After you have successfully signed in, click **Connect**.

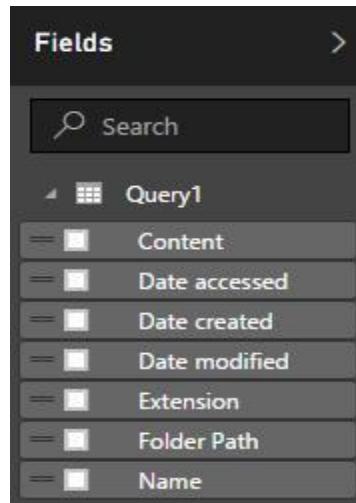


5. The next dialog box shows the file that you uploaded to your Data Lake Store account. **Verify** the info and then click **Load**.



Content	Name	Extension	Date accessed	Date modified	Date created	Folder Path
[Table]	demotd2k45		7/5/2017 11:27:43 AM	7/5/2017 11:28:14 AM		https://cddatalake2k45.azuredatalakestore.net/webhdfs/v1/

6. After the data has been successfully loaded into Power BI, you will see the available fields in the **Fields** tab.



The screenshot shows the 'Fields' pane in Power BI. At the top, there is a search bar with the placeholder 'Search'. Below the search bar, a tree view shows a single node 'Query1' expanded, revealing seven child fields: 'Content', 'Date accessed', 'Date created', 'Date modified', 'Extension', 'Folder Path', and 'Name'. Each field is represented by a small icon and a text label.

7. However, to visualize and analyze the data, you might prefer the data be available as per your requirements. To do so, follow the steps below:

8. Select **Edit Query** from the top menu bar:



Under the content column, right click on **Table** and select **Add as New Query**, you will see a new query added in the queries column:

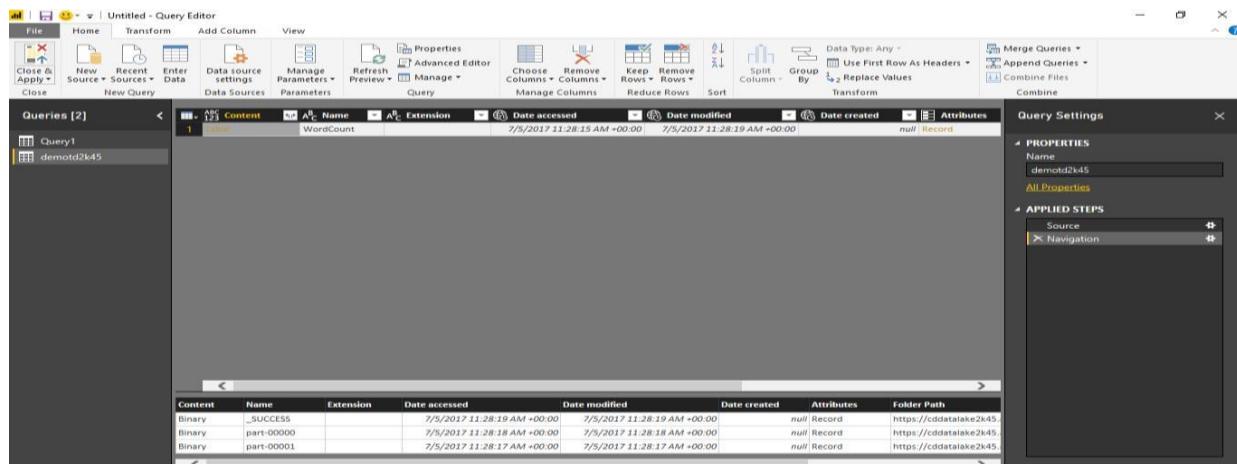


The screenshot shows the Power BI Query Editor. The 'Content' column is selected in the data grid. The ribbon at the top has a 'Table' icon highlighted with a red box. The 'Query Settings' pane on the right shows the query is named 'Query1'. The data grid displays the following table:

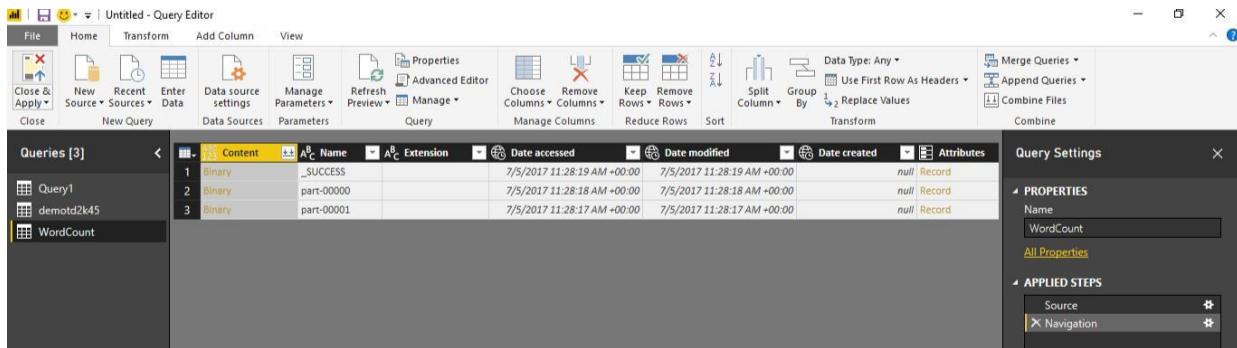
Content	Name	Extension	Date accessed	Date modified	Date created	Attributes
demoted243	demoted243	demoted243	7/3/2019 11:27:43 AM +05:30	7/3/2019 11:39:23 AM +05:30	7/3/2019 11:27:43 AM +05:30	null - Trashed



9. Once again, **right click** and select **Add as New Query** to convert the table content to binary form.



10. Right click and create a new query to get the data from the table as shown below:



The screenshot shows the Power BI Query Editor interface. The main area displays a table with the following columns and data:

	Content	Name	Extension	Date accessed	Date modified	Date created	Attributes
1	Binary	_SUCCESS		7/5/2017 11:28:19 AM +0:00	7/5/2017 11:28:19 AM +0:00	7/5/2017 11:28:19 AM +0:00	null Record
2	Binary	part-00000		7/5/2017 11:28:18 AM +0:00	7/5/2017 11:28:18 AM +0:00	7/5/2017 11:28:18 AM +0:00	null Record
3	Binary	part-00001		7/5/2017 11:28:17 AM +0:00	7/5/2017 11:28:17 AM +0:00	7/5/2017 11:28:17 AM +0:00	null Record

The Query Settings pane on the right shows the following properties:

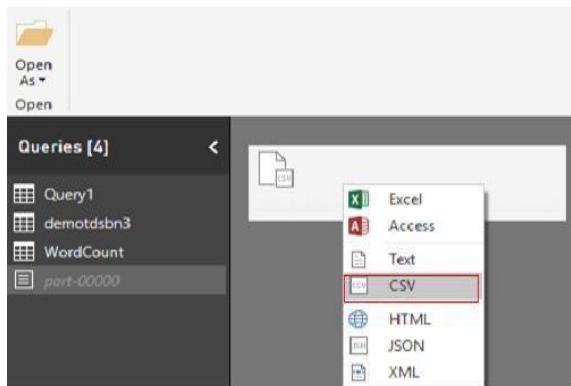
- PROPERTIES**: Name = WordCount
- APPLIED STEPS**: Source, Navigation

11. You will see a file icon that represents the file that you uploaded. Right-click the file, and click **CSV**.



The screenshot shows the Power BI Query Editor interface. The main area displays a file icon representing a CSV file. The Query Settings pane on the right shows the following properties:

- PROPERTIES**: Name = part-00000
- APPLIED STEPS**: Source, Navigation



The screenshot shows the Power BI Query Editor interface. A context menu is open over a file icon, and the 'CSV' option is highlighted. The Query Settings pane on the right shows the following properties:

- PROPERTIES**: Name = part-00000
- APPLIED STEPS**: Source, Navigation



12. Your data is now available in a format that you can use to create visualizations.

13. From the **Home** ribbon, click **Close and Apply**, and then click **Close and Apply**.

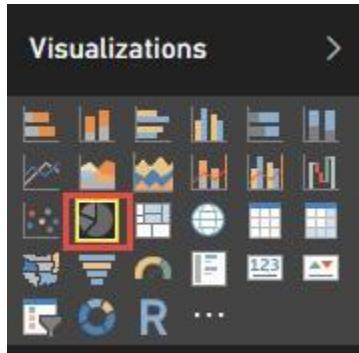


14. Once the query is updated, the **Fields** tab will show the new fields available for visualization.

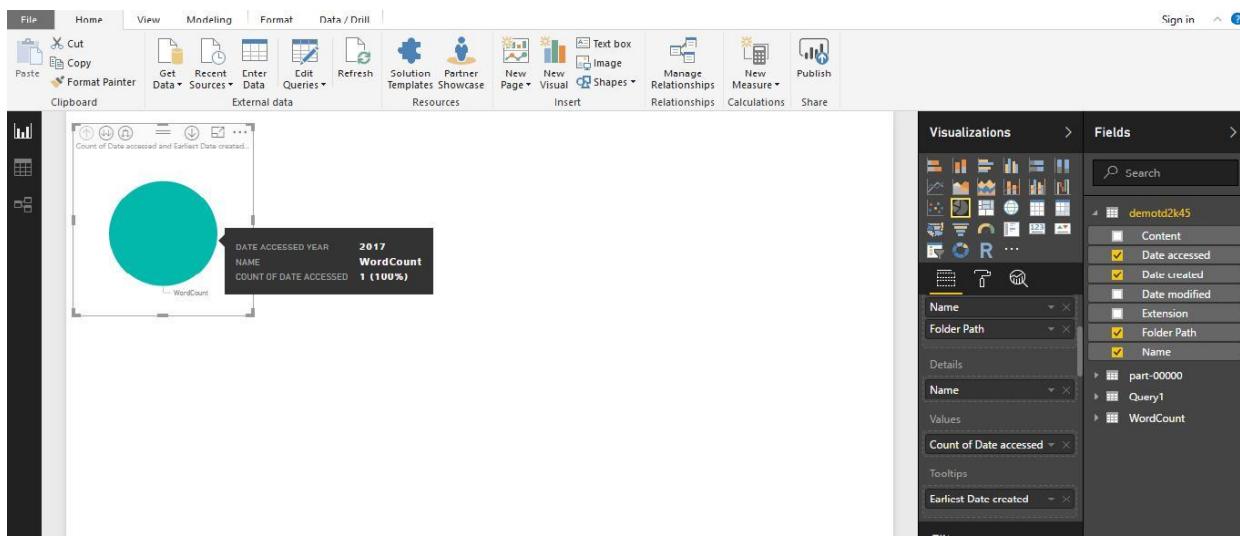


15. You can create a pie chart to represent your data. To do so, make the following selections:

- From the **Visualizations** tab, click the symbol for a **pie chart** (see below).



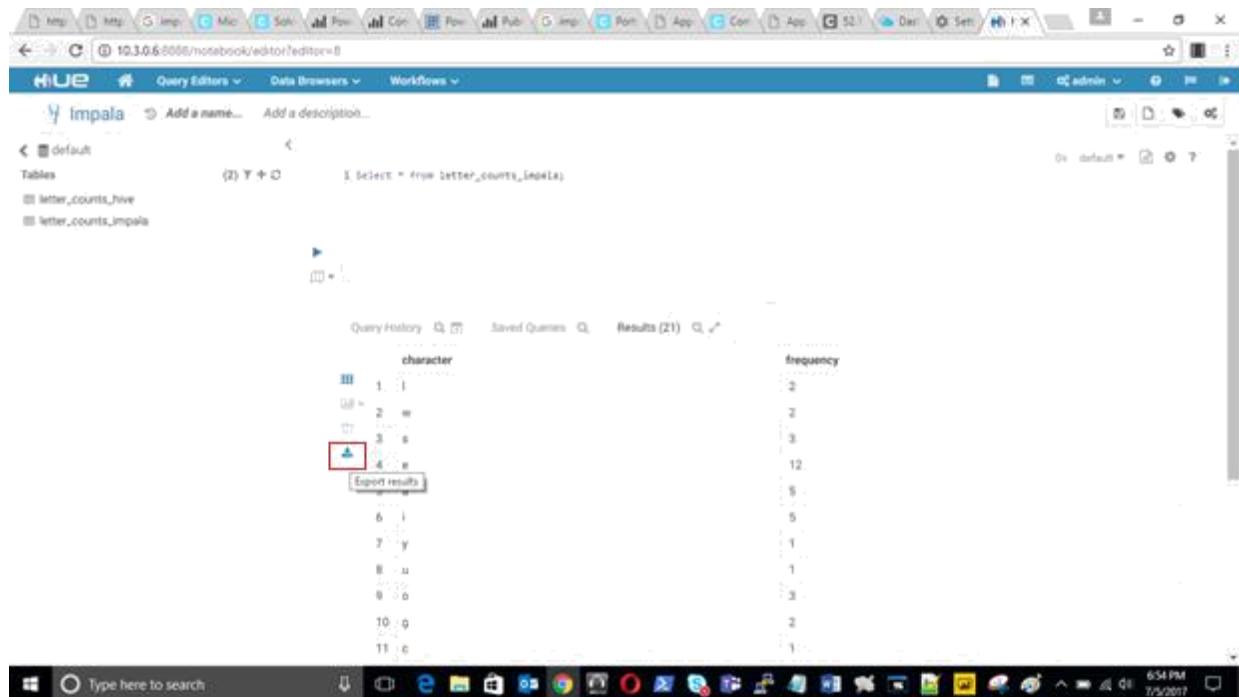
- Drag the columns that you want to use and represent in your pie-chart from the **Fields** tab to **Visualizations** tab, as shown below:



16. From the **file** menu, click **Save** to save the visualization as a Power BI Desktop file.

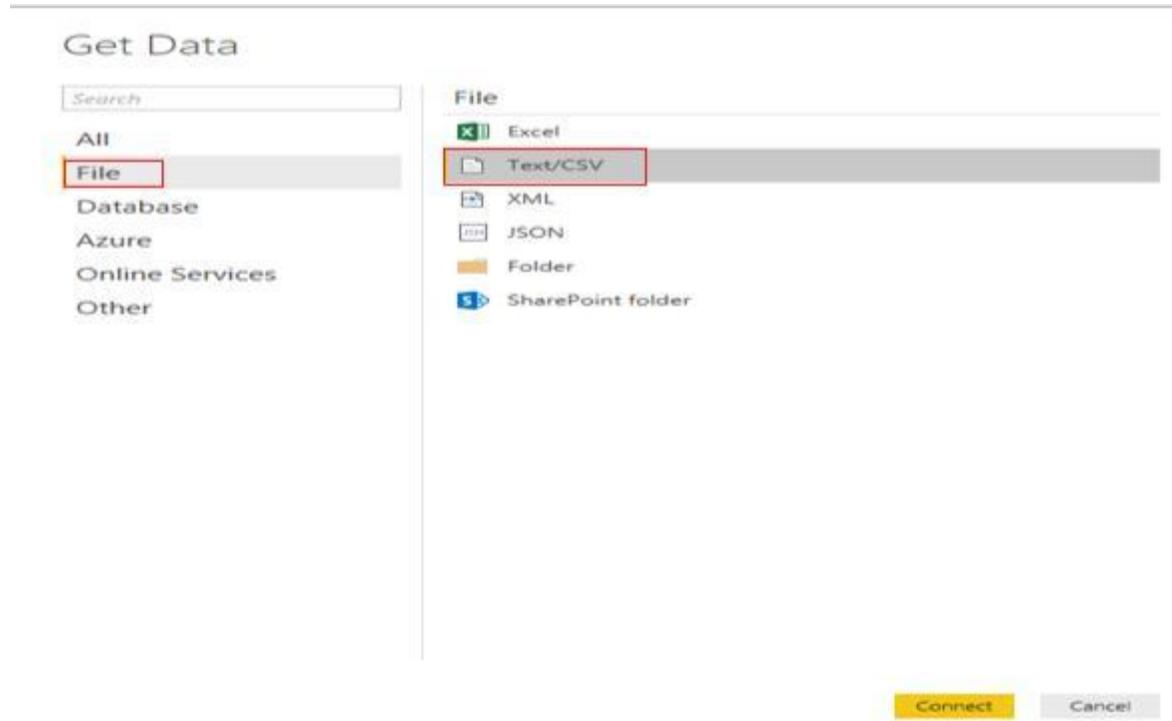
5.2 Integrating with Impala

1. Go to **point 7** of section **4.7**, where you ran a query from the table created using the output from ADLS copied to local HDFS.



2. Click the **Export Results** button in the Hue Impala UI, as seen in the above screenshot, to download the output as a **CSV** file.

3. From the Home ribbon in Power BI, click Get Data, and then click **More**. In the Get Data dialog box, click File, click Text/CSV, and then click Connect.

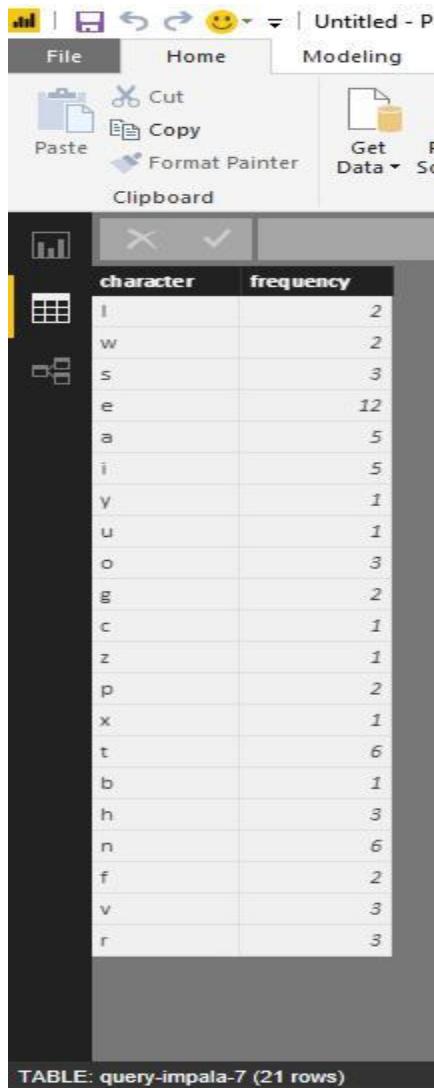


4. Select the **CSV** file exported from Impala in **Step 2** and click on **Open**.



5. Click on **Load**.

6. Select the **Data** button to visualize the content.



The screenshot shows the Power BI desktop interface with a dark theme. The ribbon at the top has 'File', 'Home', and 'Modeling' tabs. The 'Home' tab is selected, showing icons for 'Paste', 'Cut', 'Copy', 'Format Painter', 'Clipboard', and 'Get Data'. Below the ribbon is a table visualization with the following data:

character	frequency
l	2
w	2
s	3
e	12
a	5
i	5
y	1
u	1
o	3
g	2
c	1
z	1
p	2
x	1
t	6
b	1
h	3
n	6
f	2
v	3
r	3

At the bottom of the table visualization, it says 'TABLE: query-impala-7 (21 rows)'.

You have successfully visualized the content exported from impala using power BI.

5. Reference

6.1 Configure SOCKS Proxy

Please refer to below documentation from Cloudera to help setup a SOCKS Proxy.

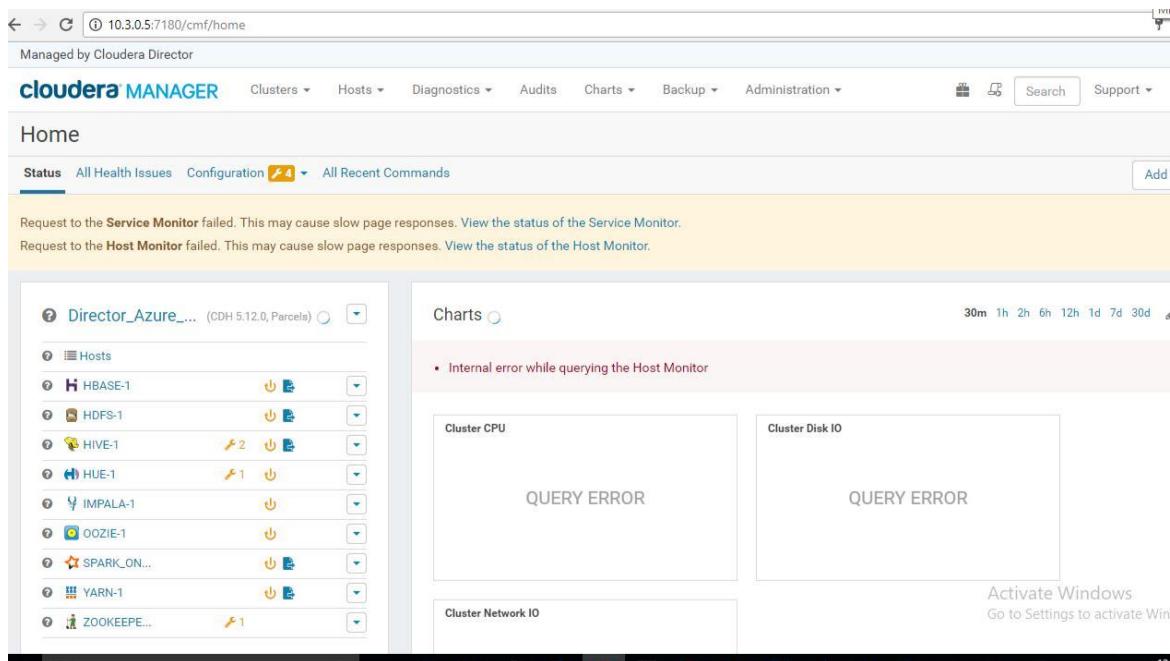
https://www.cloudera.com/documentation/director/latest/topics/director_get_started_azure_socks.html#concept_a35_k4l_zw

6.2 Restart Cloudera Management Service

You may need to restart Cloudera Management Service for the below errors:

Error:

- Request to the Service Monitor failed. This may cause slow page responses. [View the status of the Service Monitor.](#)
- Request to the Host Monitor failed. This may cause slow page responses. [View the status of the Host Monitor.](#)



The screenshot shows the Cloudera Manager Home page at the URL 10.3.0.5:7180/cmf/home. The status bar at the top displays two error messages: "Request to the Service Monitor failed. This may cause slow page responses. View the status of the Service Monitor." and "Request to the Host Monitor failed. This may cause slow page responses. View the status of the Host Monitor." The main interface includes a navigation bar with links for Clusters, Hosts, Diagnostics, Audits, Charts, Backup, and Administration. On the left, there is a list of hosts: HBASE-1, HDFS-1, HIVE-1, HUE-1, IMPALA-1, OOZIE-1, SPARKEON..., YARN-1, and ZOOKEEPER-1. On the right, there are three charts: Cluster CPU, Cluster Disk IO, and Cluster Network IO, all showing "QUERY ERROR". A message at the bottom right says "Activate Windows Go to Settings to activate Win".

Restarting Cloudera Management Service:

1. Go to `http://<Manager Node private IP Address>:7180/cmf/home`.

Example: <http://10.3.0.5:7180/cmf/home>

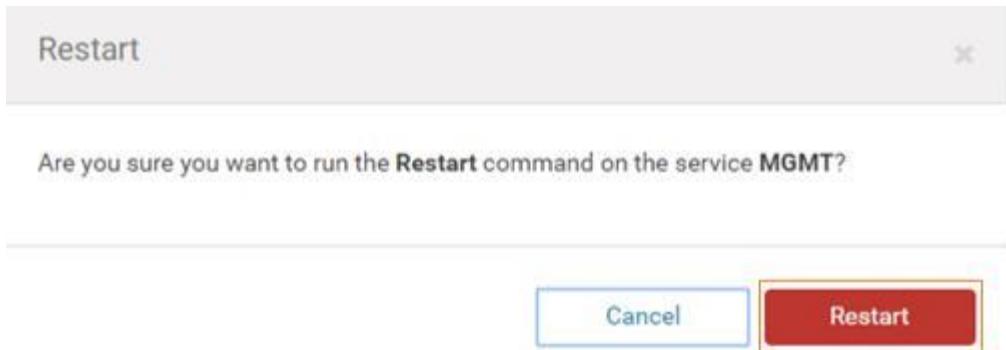
2. Go to **Cloudera Management Service** and select **MGMT**.

The screenshot shows the Cloudera Management Service interface. On the left, a sidebar lists various services: Director_Azure_..., Hosts, HBASE-1, HDFS-1, HIVE-1, HUE-1, IMPALA-1, OOZIE-1, SPARK_ON..., YARN-1, and ZOOKEEPER-1. Below this, a section titled 'Cloudera Management Service' shows a dropdown menu with 'MGMT' selected. On the right, there are two charts: 'Cluster CPU' and 'Cluster Network IO', both of which display a 'QUERY ERROR' message.

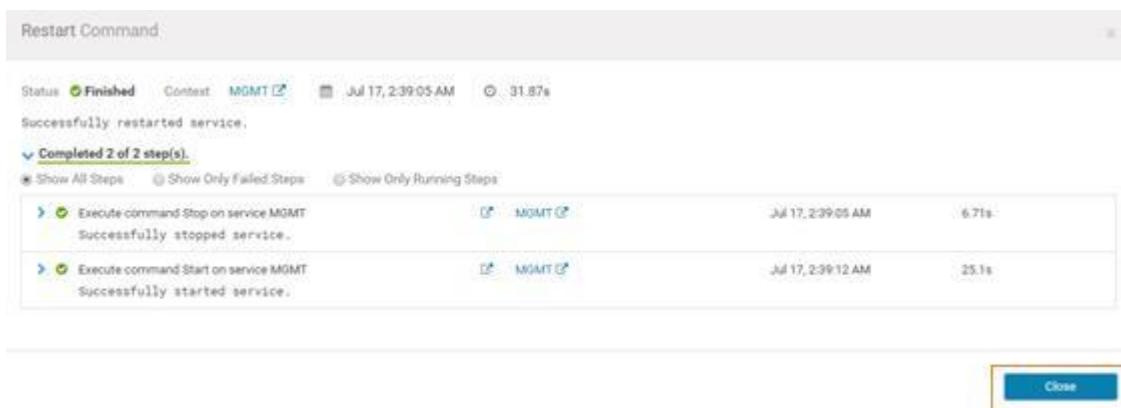
3. Click on the drop down menu and select **Restart**.

The screenshot shows the 'MGMT Actions' dropdown menu. The 'Restart' option is highlighted with a yellow border. Other options in the menu include Start, Stop, Instances, Configuration, Add Role Instances, Rename, Delete, and View Maintenance Mode Status.

4. Confirm by clicking the **Restart** button.



5. Click on **Close** to complete the process.



Note: If you performed this restart in response to errors, please now re-run section **4.3** after performing the above steps.

6.3 Error Messages While Running the Spark Job

1. You may see a few errors popping up while executing the Spark job that can safely be ignored, such as the ones below.

Note: The permissions get properly set in the .sh file.

```
sh ClouderaSparkSetup.sh demotdweti 10.3.0.6
mkdir: Permission denied: user=cloudera, access=WRITE, inode="/":hdfs:supergroup:drwxr-xr-x --
2017-07-11 16:55:54-- https://aztdrepo.blob.core.windows.net/clouderadirector/wordcount.jar
Resolving aztdrepo.blob.core.windows.net... 52.238.56.168
Connecting to aztdrepo.blob.core.windows.net|52.238.56.168|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6371588 (6.1M) [application/octet-stream]
Saving to: "/home/cloudera/wordcount.jar"
```

2. Searching for Cloudera Navigator – this error can safely be ignored.

```
INFO scheduler.DAGScheduler: Job 1 finished: saveAsTextFile at SparkWordCount.scala:32,
took 1.811055 s
INFO spark.SparkContext: Invoking stop() from shutdown hook
ERROR scheduler.LiveListenerBus: Listener ClouderaNavigatorListener threw an exception
java.io.FileNotFoundException: Lineage is enabled but lineage directory
/var/log/spark/lineage doesn't exist
at
com.cloudera.spark.lineage.ClouderaNavigatorListener.checkLineageEnabled(ClouderaNavigatorListener.scala:122)
at com.cloudera.spark.lineage.
```

Note: You may refer to the **Spark** section of the **Cloudera release notes** for further details (link below).

https://www.cloudera.com/documentation/enterprise/releasenotes/topics/cn_rn_known_issues.html