

HIVE TUTORIAL

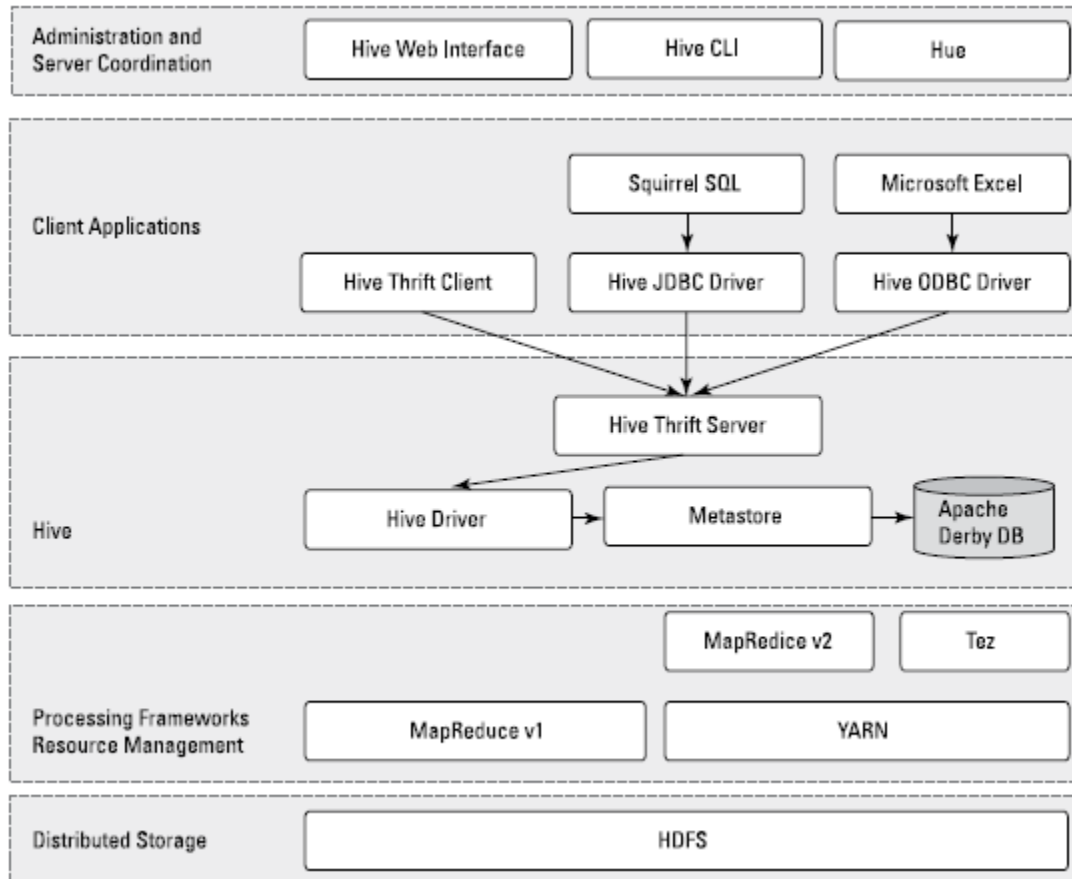
Hands-on Session

by Suchitra Jayaprakash
suchitra@cmi.ac.in

Apache HIVE

- Apache Hive was created at Facebook by a team of engineers led by Jeff Hammerbacher.
- HIVE uses SQL dialect known as HIVE QUERY LANGUAGE (HiveQL).
- HIVE hides the complexity of MapReduce. It provides SQL type script to perform MapReduce task.
- HIVE is data warehouse for managing and processing structured data.
- Hive supports "**READ Many WRITE Once**" pattern. Hive is "**Schema on READ only**".

Apache Hive Architecture



(source: Hadoop for Dummies)

Run HIVE

- **Start Cloudera server**

```
docker run --hostname=quickstart.cloudera --privileged=true -t -i --  
publish-all=true -p 8888:8888 -p 8080:80 -p 50070:50070 -p 8088:8088 -p  
50075:50075 -p 8032:8032 -p 8042:8042 -p 19888:19888 -p 10000:10000  
cloudera/quickstart /usr/bin/docker-quickstart
```

- **To get HIVE command prompt**
- Type hive and press enter to get hive command line interface.

HIVE CLI

```
Logging initialized using configuration in fi
properties
WARNING: Hive CLI is deprecated and migration
hive>
>
> CREATE SCHEMA user_db;
OK
Time taken: 3.927 seconds
hive> show databases;
OK
default
user_db
Time taken: 1.808 seconds, Fetched: 2 row(s)
hive>
```

- Create Database Statement:

CREATE SCHEMA <database name>;

It creates database in hive. Database is collection of table.

SHOW DATABASES;

It displays the list of databases in hive instance.

- Create Table Statement:

```
CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS] [db_name.] table_name
[(col_name data_type , ...)]
[COMMENT table_comment]
[ROW FORMAT row_format]
[STORED AS file_format]
```

HiveQL

- Load OnlineStore dataset into HDFS using HIVE and perform few aggregation operations:

Step 1 – Create Table

```
CREATE TABLE IF NOT EXISTS Online_Retail ( InvoiceNo STRING, StockCode STRING, Description  
STRING,Quantity INT, InvoiceDate TIMESTAMP, UnitPrice double ,CustomerID INT,Country STRING )  
COMMENT 'Online Retail Data Set'  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY ','  
LINES TERMINATED BY '\n'  
STORED AS TEXTFILE;
```

```
CREATE TABLE IF NOT EXISTS tmp(InvoiceNo STRING, StockCode STRING, Description STRING,  
Quantity INT, InvoiceDate STRING,UnitPrice double ,CustomerID STRING,Country STRING)  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY ','  
LINES TERMINATED BY '\n'  
STORED AS TEXTFILE;
```

```
hive> CREATE TABLE IF NOT EXISTS Online_Retail ( InvoiceNo STRING, StockCode STR  
ING, Description STRING,  
> Quantity INT, InvoiceDate TIMESTAMP,UnitPrice double ,CustomerID INT,Count  
ry STRING )  
> COMMENT 'Online Retail Data Set'  
> ROW FORMAT DELIMITED  
> FIELDS TERMINATED BY ','  
> LINES TERMINATED BY '\n'  
> STORED AS TEXTFILE;  
OK  
Time taken: 2.193 seconds  
hive> CREATE TABLE IF NOT EXISTS tmp(InvoiceNo STRING, StockCode STRING, Descrip  
tion STRING,  
> Quantity INT, InvoiceDate STRING,UnitPrice double ,CustomerID STRING,Count  
ry STRING)  
> ROW FORMAT DELIMITED  
> FIELDS TERMINATED BY ','  
> LINES TERMINATED BY '\n'  
> STORED AS TEXTFILE;  
OK  
Time taken: 0.417 seconds  
hive>
```

HiveQL

Step 2 – Load Data

- Copy Text file to docker container

```
docker cp E:/MSc_Datascience/BigDataHadoop/Slides/hive/Online_Retail.csv  
<containerid>:/tmp/Online_Retail.csv
```

- Load Hive tables

```
LOAD DATA LOCAL INPATH '/tmp/Online_Retail.csv'  
OVERWRITE INTO TABLE tmp;
```

```
INSERT INTO TABLE Online_Retail  
SELECT InvoiceNo, StockCode, Description, Quantity,  
from_unixtime(unix_timestamp(InvoiceDate, 'dd-MM-yyyy HH:mm')),  
UnitPrice, CustomerID, Country  
FROM tmp;
```

HiveQL

```
Time taken: 0.117 seconds
hive> LOAD DATA LOCAL INPATH '/tmp/Online_Retail.csv'
> OVERWRITE INTO TABLE tmp;
Loading data to table default.tmp
Table default.tmp stats: [numFiles=1, numRows=0, totalSize=46123538, rawDataSize=0]
OK
Time taken: 8.551 seconds
hive>
```

```
hive> INSERT INTO TABLE Online_Retail
> SELECT InvoiceNo, StockCode, Description, Quantity,
> from_unixtime(unix_timestamp(InvoiceDate, 'dd-MM-yyyy HH:mm')),
> UnitPrice, CustomerID, Country
> FROM tmp;
Query ID = root_20200314175050_d13e5e65-f5bd-4c27-a85d-9fc45fe129a9
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1584207403190_0001, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1584207403190_0001/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1584207403190_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2020-03-14 18:00:16,234 Stage-1 map = 0%, reduce = 0%
2020-03-14 18:01:17,174 Stage-1 map = 0%, reduce = 0%, Cumulative CPU 15.73 sec
2020-03-14 18:02:17,926 Stage-1 map = 0%, reduce = 0%, Cumulative CPU 53.73 sec
2020-03-14 18:02:24,806 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 58.01 sec
MapReduce Total cumulative CPU time: 58 seconds 10 msec
Ended Job = job_1584207403190_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to: hdfs://quickstart.cloudera:8020/user/hive/warehouse/online_retail_1/hive-staging_hive_2020-03-14_17-58-35_519_7469576875740326583-1/-ext-10000
Loading data to table default.online_retail
Table default.online_retail stats: [numFiles=1, numRows=541909, totalSize=47486522, rawDataSize=46944613]
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Cumulative CPU: 58.01 sec HDFS Read: 46128029 HDFS Write: 47486609 SUCCESS
Total MapReduce CPU Time Spent: 58 seconds 10 msec
OK
Time taken: 240.053 seconds
hive>
```


HiveQL

Step 3 – Select operation

SELECT * from Online_Retail LIMIT 5;

```
hive> SELECT * from Online_Retail LIMIT 5;
OK
536365 85123A WHITE HANGING HEART T-LIGHT HOLDER      6      2010-12-01 08:26:00
:00    2.55 17850 United Kingdom
536365 71053  WHITE METAL LANTERN        6      2010-12-01 08:26:00    3.39 1
7850   United Kingdom
536365 84406B CREAM CUPID HEARTS COAT HANGER    8      2010-12-01 08:26:00    2
.75    17850 United Kingdom
536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE  6      2010-12-01 08:26:00
:00    3.39 17850 United Kingdom
536365 84029E RED WOOLLY HOTTIE WHITE HEART.    6      2010-12-01 08:26:00    3
.39    17850 United Kingdom
Time taken: 0.355 seconds, Fetched: 5 row(s)
hive>
```

SELECT InvoiceDate from Online_Retail LIMIT 5;

SELECT Country,count(*) FROM Online_Retail GROUP BY Country;

SELECT * FROM Online_Retail WHERE UnitPrice>1000 AND Country = 'United Kingdom';

Drop table

DROP TABLE IF EXISTS tmp;

THANK YOU