# Management and Processing of Big Data

## Level-I

## session-4

# Recap

- Hadoop Installation
- Sqoop

# Agenda for today

- Data warehousing using Hive/Impala
  - HQL Commands
  - Data ingestion process
  - Integrity checks in Hive tables
  - ETL
  - Run Hive/Impala scripts
  - Export data from Impala table to local file system
  - Connectivity with RDBMS

# Commands

- SHOW
- CREATE
- DROP
- ALTER
- SELECT
- INSERT
- INSERT OVERWRITE

# SHOW

- SHOW DATABASES

- SHOW TABLES

- SHOW CREATE TABLE movie;

# CREATE

- CREATE DATABASE mydb;
- CREATE DATABASE mydb LOCATION '/data/mydb’;
- CREATE TABLE movie (movieId INT, movieName STRING);
- CREATE TABLE movie (movieId INT, movieName STRING) LOCATION '/data/mydb/movie' ;
- CREATE TABLE movie (movieId INT, movieName STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘,’;
- CREATE TABLE movie_stage LIKE movie;
- CREATE TABLE movie_stage AS SELECT * FROM movie LIMIT 0;

# DROP

- DROP DATABASE mydb

- DROP TABLE movie

# ALTER

- ALTER TABLE movie SET LOCATION ‘/data/mydb/movie2’

- ALTER TABLE movie SET TBLPROPERTIES ('field.delim'=",")

- ALTER TABLE movie CHANGE movieId movieId DECIMAL(12,0);

# SELECT

- SELECT * FROM movie t LEFT JOIN movie_stage s ON t.movieId = s.movieId
- SELECT * FROM movie t RIGHT JOIN movie_stage s ON t.movieId = s.movieId
- SELECT * FROM movie t JOIN movie_stage s ON t.movieId = s.movieId
- SELECT * FROM movie t FULL OUTER JOIN movie_stage s ON t.movieId = s.movieId

# INSERT

- INSERT INTO TABLE movie VALUES (1,”Avengers”)
- INSERT INTO TABLE movie SELECT * FROM movie_stage
- INSERT INTO TABLE movie VALUES (1,”Avengers”), (2,”Ironman”)
- INSERT INTO TABLE movie (movieName, movieId) SELECT movieName, movieId FROM movie_stage

# INSERT OVERWRITE: update/delete

- INSERT OVERWRITE TABLE movie_stage SELECT movieId, CASE movieId WHEN 1 THEN ‘Transformers’ ELSE movieName END FROM movie_stage

- INSERT OVERWRITE TABLE movie_stage SELECT * FROM movie_stage WHERE movieId != null;

- INSERT OVERWRITE TABLE movie_stage SELECT * FROM movie_stage LIMIT 0;

# Miscellaneous

- DESCRIBE movie

- DESCRIBE FORMATTED movie

- TRUNCATE TABLE movie

- LOAD DATA INPATH ‘/hdfs_location/’ INTO TABLE movie

- LOAD DATA LOCAL INPATH ‘/local_location/’ INTO TABLE movie

# Practise Exercise 1

# Data Ingestion

- Ingesting source file
  - Optionally, remove header record if present using sed command as
    *sed –i ‘1d’ source_file_with_header.csv*
  - Get the table location from describe formatted
  - Copy source file to that location using copyFromLocal command
  - Gzip the source file on local system
  - Move compressed file to Archival location on HDFS

# Integrity checks

- Make sure about duplicates, null and unique violation while ingesting the file as well as populating final target table

- You may check in following order
  - Unique Primary Index (UPI)
  - NULL
  - Absolute duplicates

- You may prefer to automate this validation process

# Run Commands as Script

- hive –f script_file_name.hql
- hive –d default_db –f script_file_name.hql
- hive –e “HQL”

- impala-shell –f script_file_name.hql
- impala-shell –q “HQL”
- impala-shell –d default_db –f script_file_name.hql
- impala-shell –o output_file.csv script_file_name.hql
- impala-shell --output_delimiter=‘,’ –o output_file.csv –f script_file_name.hql

# Export Data to Local File Storage

- Export whole table

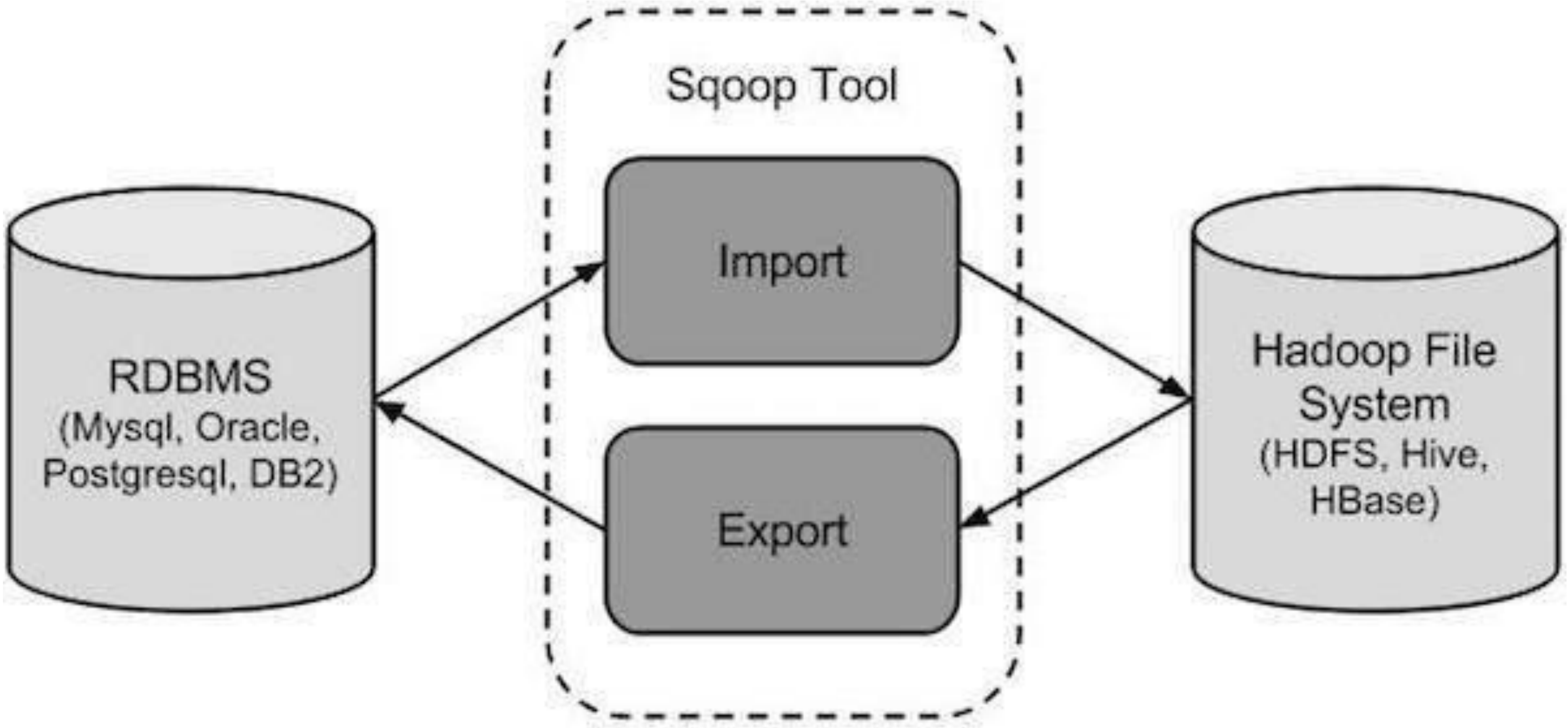
hadoop fs -getmerge <HDFS Path> <Local Path>

- Use –o {output_file} option of impala-shell to export ad-hoc data by joining multiple tables

# Practise Exercise 2

# Connectivity with RDBMS

- SQOOP



# Sqoop Export

```
sqoop export --connect jdbc:mysql://localhost:3306/<DB name>
             --username <User name>
             --password <Password>
             --table <Table name>
             --fields-terminated-by ‘,’
             --export-dir <HDFS directory name>
```

# Sqoop Import

```
sqoop import --connect jdbc:mysql://localhost:3306/retail_db
                    --username retail_dba
                    -p
                    --table test
                    --m 1
                    --target-dir /data/mydb/test
```

# Sqoop: How to handle null?

- --null-string '\N’

- --null-non-string '\N’

# References

- Reference Book
  - Hadoop: The definitive guide by Tom White (Weblink)
- Impala SQL guide
  - https://www.cloudera.com/documentation/enterprise/5-8-x/topics/impala_langref_sql.html
- Impala shell options
  - https://www.cloudera.com/documentation/enterprise/5-9-x/topics/impala_shell_options.html
- Sqoop User’s guide
  - https://sqoop.apache.org/docs/1.4.6/SqoopUserGuide.html