Lab: Data Ingestion With Sqoop

Using Sqoop To Ingest RDBMS Data

Lab: Data Ingestion With Sqoop

1. Table of Contents

2. Ingesting RDBMS Table With Sqoop

The goal of this lab is to demonstrate the process of ingesting RDBMS data using Sqoop

Before we start our ingestion, lets prepare some data into postgresql

1. Login to Cockpit and ssh to root@master1.cluster

```
ssh root@master1.cluster
```

2. Download dataset

```
wget http://repo.kagesenshi.org/hdptraining/weather_data.csv.gz
gzip -d weather_data.csv.gz
mv weather_data.csv /tmp/
```

3. Switch to postgres user

```
su - postgres
```

4. Create database 'weather'

```
createdb weather
```

5. Launch postgresql shell

```
psql -d weather
```

6. Create table and import dataset

```
create table weather_data (
    station_identifier character varying(50),
    measurement_date date,
    measurement_type character varying(50),
    measurement_flag integer,
    quality_flag character varying(2),
    source_flag character varying(2),
    observation_time character varying(2),
    notes character varying(50)
);

copy weather_data(station_identifier, measurement_date, measurement_type,
measurement_flag, quality_flag, source_flag, observation_time, notes) from
'/tmp/weather_data.csv' delimiter ',' csv;

select * from weather_data limit 10;
```

Lab: Data Ingestion With Sqoop

2.1. Ingesting Using Normal JDBC Connection

Now lets ingest this dataset.

1. Login to Cockpit and ssh to root@edge.cluster

```
ssh root@edge.cluster
```

2. Install postgresql JDBC driver

```
yum install postgresql-jdbc -y
cp /usr/share/java/postgresql-jdbc.jar /usr/hdp/current/sqoop-client/lib/
```

3. Switch to user admin

```
su - admin
```

4. Lets connect to postgresql in master1 and list available tables using Sqoop

```
sqoop list-tables --connect jdbc:postgresql://master1.cluster:5432/weather --
username postgres
```

5. Lets ingest this table into ORC weather_data table in Hive (Note: you may want to stop HiveServer2 Interactive for this to work in the VM due to limited YARN memory)

```
sqoop import \
    --connect jdbc:postgresql://master1.cluster:5432/weather \
    --username postgres \
    --table weather_data \
    --create-hcatalog-table \
    --hcatalog-database default \
    --hcatalog-table weather_data \
    --hcatalog-storage-stanza "stored as orc" \
    -m 2
    --split-by measurement_date
```

6. Login to Ambari and navigate to Resource Manager UI to view Sqoop job details.
7. Connect to Hive (use port 10000 if HS2 Interactive is stopped)

```
beeline -u jdbc:hive2://master1.cluster:10500/default -n admin
```

8. Lets check the table

```
SELECT * FROM weather_data LIMIT 10;
```

2.2. Ingesting Using Sqoop Direct Mode

1. Login to Cockpit and ssh to root@edge.cluster

```
ssh root@edge.cluster
```

2. PostgreSQL direct mode ingestion requires psql command to be available. Lets install postgresql client package

   yum install postgresql -y

3. Switch to user admin

```
su - admin
```

4. Lets ingest this table into ORC weather_data_direct table in Hive using direct mode (Note: PostgreSQL direct mode does not support ingestion straight into Hive) (Note: You may want to stop HiveServer2 Interactive for this to work in the VM due to limited YARN memory)

```
sqoop import \
    --connect jdbc:postgresql://master1.cluster:5432/weather \
    --username postgres \
    --table weather_data \
    --target-dir /tmp/weather_data/ \
    --direct
```

5. Connect to Hive (use port 10000 if HS2 Interactive is stopped)

```
beeline -u jdbc:hive2://master1.cluster:10500/default -n admin
```

6. Lets import the table

```
CREATE TEMPORARY EXTERNAL TABLE weather_data_tmp(
  station_identifier varchar(50),
  measurement_date string,
  measurement_type varchar(50),
  measurement_flag int,
  quality_flag varchar(2),
  source_flag varchar(2),
  observation_time varchar(2),
  notes varchar(50))
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
STORED AS TEXTFILE
LOCATION  '/tmp/weather_data';

CREATE TABLE weather_data_direct STORED AS ORC AS SELECT * FROM weather_data_tmp;
```

7. Lets check the table

```
SELECT * FROM weather_data_direct LIMIT 10;
```