Peer-Graded Assignment: Data Management
Course: Managing Big Data in Clusters and Cloud Storage
Name: Theodor-Mihai Iliant
Date: 13th January 2021

Assignment

Create a table named **tbm_sf_la** in the database named **dig** to store the data from three tunnel boring machines (TBMs), which is currently stored in S3 in three separate subdirectories under a directory named **tbm_sf_la** in the bucket named **training-coursera2**. In this document, describe the steps taken to complete this task.

Solution

I performed the following steps to complete this task:

1. To examine the data, I used the following piped commands on the terminal:
   hdfs dfs -cat s3a://training-coursera2/tbm_sf_la/central/hourly_central.csv | head -n 5

   hdfs dfs -cat s3a://training-coursera2/tbm_sf_la/north/hourly_north.csv | head -n 5

   hdfs dfs -cat s3a://training-coursera2/tbm_sf_la/south/hourly_south.tsv | head -n 5

2. To create the three separate tables and load data into them, I used the following SQL commands :

For creating tables:

```
create table dig.central
    (tbm string,year char(4), month char(2), day char(2), hour char(2),
    dist decimal(8,2), lon decimal(9,6), lat decimal(9,6))
    row format delimited
    fields terminated by ','
    tblproperties('serialization.null.format' = '999999', 'skip.header.line.count' = '1');
```

```
create table dig.north
    (tbm string,year char(4), month char(2), day char(2), hour char(2),
    dist decimal(8,2), lon decimal(9,6), lat decimal(9,6))
    row format delimited
    fields terminated by ','
    tblproperties('serialization.null.format' = '\N');
```

```
create table dig.south
    (tbm string,year char(4), month char(2), day char(2), hour char(2),
    dist decimal(8,2), lon decimal(9,6), lat decimal(9,6))
    row format delimited
    fields terminated by '\t'
    tblproperties('serialization.null.format' = '\N');
```

For loading data from the Amazon server s3a://training_coursera2/tbm_sf_la into each of the three tables, I used the following commands on the terminal:

hdfs dfs -cp s3a://training-coursera2/tbm_sf_la/central/hourly_central.csv hdfs:///user/hive/warehouse/dig.db/central

hdfs dfs -cp s3a://training-coursera2/tbm_sf_la/north/hourly_north.csv hdfs:///user/hive/warehouse/dig.db/north

hdfs dfs -cp s3a://training-coursera2/tbm_sf_la/south/hourly_south.tsv hdfs:///user/hive/warehouse/dig.db/south

I handled missing values using serialization.null.format as it can be seen in the photos on previous page. For example, for 999999, use tblproperties('serialization.null.format' = '999999').

3. To create the tbm_sf_la table, I performed a CTAS statement - I did this using two UNION clauses because the three tables have the same number of columns and the names of the columns are also the same - aka they have the same structure/schema:

```
create table tbm_sf_la as
    select * from central
    union all
    select * from north
    union all
    select * from south
```

! The above query was run using the dig database as the active database, so that the tbm_sf_la is stored in the directory ../dig/ .

Result

SELECT tbm, COUNT(*) AS num_rows FROM dig.tbm_sf_la GROUP BY tbm ORDER BY tbm;

tbm	num_rows
Bertha II	91619
Diggy McDigface	93163
Shai-Hulud	94237

DESCRIBE dig.tbm_sf_la;

name	type
tbm	string
year	char(4)
month	char(2)
day	char(2)
hour	char(2)
dist	decimal(8,2)
lon	decimal(9,6)
lat	decimal(9,6)

Notes:

For better efficiency, partition the table by the column 'tbm'.