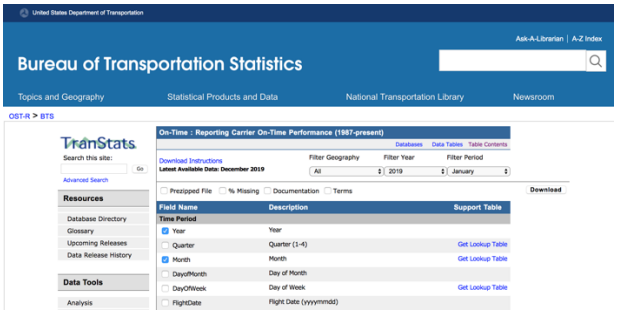
WEEK 7 ASSIGNMENT 2

Large-Scale Data Storage Systems – DATA-5400 | Spring 2020

Christina Morgenstern

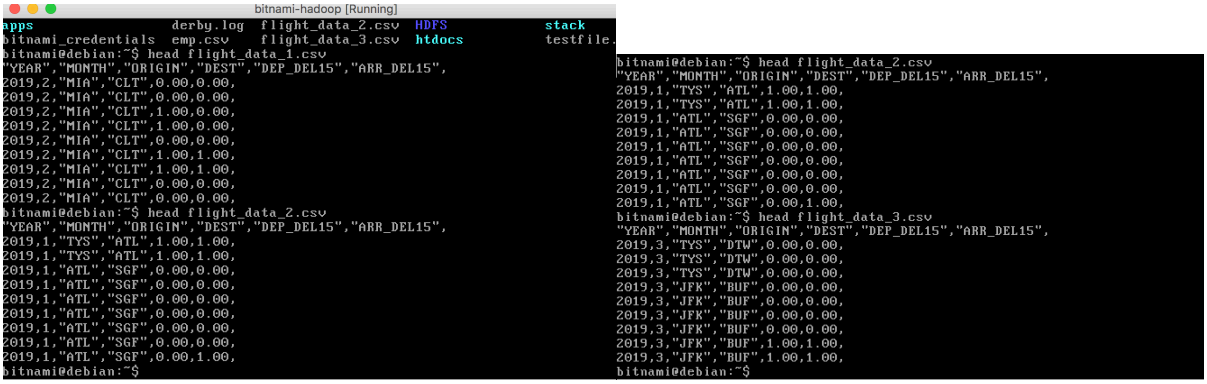
The data for assignment 2 was downloaded from the US Department of Transportation – Bureau of Transportation Statistics and comprises of Airline On-Time performance data for the first three months of the year 2019.
The following columns were selected for download: Origin, Dest(ination), DepDel15 (Departure Delay), ArrDel15 (Arrival Delay). For every month, you need to do a separate download.



The three files with the respective data for January, February and March 2019 were downloaded to my machine, unzipped and renamed to flight_data_1.csv, flight_data_2.csv and flight_data_3.csv.
Using SCP the three files were transferred to my Bitnami Hadoop Virtual Machine running via VirtualBox. SSH was enabled on the VM beforehand as previously described.

```
Schreibtisch — -bash — 80×24
cp: bitnami@10.0.0.14: No such file or directory
(base) Christinas-MacBook-Pro:Desktop Christina$ scp flight_data_1.csv bitnami@1
0.0.0.14:/home/bitnami
ssh: connect to host 10.0.0.14 port 22: Connection refused
lost connection
(base) Christinas-MacBook-Pro:Desktop Christina$ scp flight_data_1.csv bitnami@1
0.0.0.14:/home/bitnami
ssh: connect to host 10.0.0.14 port 22: Connection refused
lost connection
(base) Christinas-MacBook-Pro:Desktop Christina$ scp flight_data_1.csv bitnami@1
0.0.0.14:/home/bitnami
bitnami@10.0.0.14's password:
flight_data_1.csv                             100%   15MB  33.5MB/s   00:00
(base) Christinas-MacBook-Pro:Desktop Christina$ scp flight_data_2.csv bitnami@1
0.0.0.14:/home/bitnami
bitnami@10.0.0.14's password:
flight_data_2.csv                             100%   17MB  37.0MB/s   00:00
(base) Christinas-MacBook-Pro:Desktop Christina$ scp flight_data_3.csv bitnami@1
0.0.0.14:/home/bitnami
bitnami@10.0.0.14's password:
Permission denied, please try again.
bitnami@10.0.0.14's password:
flight_data_3.csv                             100%   18MB  24.1MB/s   00:00
(base) Christinas-MacBook-Pro:Desktop Christina$
```

Using the `head` command, I checked the contents of each file.



I realized, that I have somehow mixed up the files for January and February during naming. I renamed the files again using the `mv` command.

```
mv flight_data_1.csv feb.csv
mv flight_data_2.csv jan.csv
mv flight_data_3.csv march.csv
```

Since we need to combine the three files into one, the February and March files need to have their headers removed.

```
bitnami@debian:~$ sed '1d' feb.csv > feb-noheader.csv
bitnami@debian:~$ sed '1d' march.csv > march-noheader.csv
bitnami@debian:~$
```

Using the `cat` command, the three files were combined into one, called Q12019.csv. The `ls -lah *.csv` command shows all my csv files including the ones generated in previous assignments. I checked the file size of Q12019.csv which was 50M and seemed reasonable from the merge of the three.

```
bitnami@debian:~$
bitnami@debian:~$ cat jan.csv feb-noheader.csv march-noheader.csv > Q12019.csv
bitnami@debian:~$ ls -lah *.csv
-rw-r--r-- 1 bitnami bitnami 216K Feb 28 22:37 AllstarFull.csv
-rw-r--r-- 1 bitnami bitnami   34 Feb 27 21:20 dept.csv
-rw-r--r-- 1 bitnami bitnami    0 Feb 27 22:01 emp.csv
-rw-r--r-- 1 bitnami bitnami  17M Mar  2 20:35 feb.csv
-rw-r--r-- 1 bitnami bitnami  17M Mar  2 20:43 feb-noheader.csv
-rw-r--r-- 1 bitnami bitnami 103K Feb 28 22:08 HallOfFame.csv
-rw-r--r-- 1 bitnami bitnami  16M Mar  2 20:35 jan.csv
-rw-r--r-- 1 bitnami bitnami  18M Mar  2 20:36 march.csv
-rw-r--r-- 1 bitnami bitnami  18M Mar  2 20:43 march-noheader.csv
-rw-r--r-- 1 bitnami bitnami 954K Feb 28 20:43 People.csv
-rw-r--r-- 1 bitnami bitnami  50M Mar  2 20:46 Q12019.csv
bitnami@debian:~$
```

The `head` command lists the first 10 rows of the Q12019.csv table.

```
bitnami@debian:~$ head Q12019.csv
"YEAR","MONTH","ORIGIN","DEST","DEP_DEL15","ARR_DEL15",
2019,1,"TYS","ATL",1.00,1.00,
2019,1,"TYS","ATL",1.00,1.00,
2019,1,"ATL","SGF",0.00,0.00,
2019,1,"ATL","SGF",0.00,0.00,
2019,1,"ATL","SGF",0.00,0.00,
2019,1,"ATL","SGF",0.00,0.00,
2019,1,"ATL","SGF",0.00,0.00,
2019,1,"ATL","SGF",0.00,0.00,
2019,1,"ATL","SGF",0.00,1.00,
bitnami@debian:~$ _
```

The tail command displays the last 10 rows of the Q12019.csv file.

```
bitnami@debian:~$ tail Q12019.csv
2019,3,"ATL","ROA",0.00,0.00,
2019,3,"ROA","ATL",0.00,0.00,
2019,3,"FAT","SLC",0.00,0.00,
2019,3,"SLC","FAT",0.00,0.00,
2019,3,"BOI","SLC",0.00,0.00,
2019,3,"SLC","BOI",0.00,0.00,
2019,3,"ASE","LAX",0.00,0.00,
2019,3,"FAT","SLC",1.00,1.00,
2019,3,"ATL","ROA",0.00,0.00,
2019,3,"ROA","ATL",0.00,0.00,
bitnami@debian:~$ _
```

Log into Hive

```
bitnami@debian:~$ sudo hive
```

Create four different tables which can store the same data file Q12019 but in different formats, as textfile, ORC, Avro and Parquet.

```
hive> create table Q1_text(year int, month int, origin string, dest string, del1
5 string, arr15 string)
    > row format delimited fields terminated by ',' stored as textfile;
OK
Time taken: 1.841 seconds
hive> create table Q1_parquet(year int, month int, origin string, dest string, d
el15 string, arr15 string)
    > row format delimited fields terminated by ',' stored as parquet;
OK
Time taken: 0.264 seconds
hive> create table Q1_orc(year int, month int, origin string, dest string, del15
 string, arr15 string)
    > row format delimited fields terminated by ',' stored as orc;
OK
Time taken: 0.273 seconds
hive> create table Q1_avro(year int, month int, origin string, dest string, del1
5 string, arr15 string)
    > row format delimited fields terminated by ',' stored as avro;
OK
Time taken: 0.455 seconds
hive> _
```

Display the created tables, q1_avro, q1_orc, q1_parquet, q1_text. The other tables shown were created in previous assignments.

```
Time taken: 0.455 seconds
hive> show tables;
OK
allstar
departments
emp
fame
people
q1_avro
q1_orc
q1_parquet
q1_text
salaries
Time taken: 0.088 seconds, Fetched: 10 row(s)
hive>
```

Load the airline data stored in Q12019.csv on HDFS into the created tables with different file format.

```
hive> load data local inpath '/home/bitnami/Q12019.csv' overwrite into table q1_
text;
Loading data to table default.q1_text
OK
Time taken: 2.822 seconds
hive>
```

Use the command above to load the data into the other tables.

```
load data local inpath 'home/bitnami/Q12019.csv' overwrite into table q1_avro;
load data local inpath 'home/bitnami/Q12019.csv' overwrite into table q1_orc;
load data local inpath 'home/bitnami/Q12019.csv' overwrite into table q1_parquet;
```

Go to Bitnami and check the file sizes. Navigate to the user/hive/warehouse directory.

```
bitnami@debian:~$ hdfs dfs -ls /user
Found 2 items
drwxr-xr-x   - hadoop supergroup          0 2020-02-16 21:11 /user/hadoop
drwxr-xr-x   - hadoop supergroup          0 2020-01-22 08:49 /user/hive
bitnami@debian:~$ _
```

```
bitnami@debian:~$ hdfs dfs -ls /user/hive/warehouse
Found 9 items
drwxr-xr-x   - hadoop supergroup          0 2020-02-28 22:42 /user/hive/warehous
e/allstar
drwxr-xr-x   - hadoop supergroup          0 2020-02-28 17:24 /user/hive/warehous
e/departments
drwxr-xr-x   - hadoop supergroup          0 2020-02-28 22:18 /user/hive/warehous
e/fame
drwxr-xr-x   - hadoop supergroup          0 2020-02-28 20:49 /user/hive/warehous
e/people
drwxr-xr-x   - hadoop supergroup          0 2020-03-02 21:16 /user/hive/warehous
e/q1_avro
drwxr-xr-x   - hadoop supergroup          0 2020-03-02 21:15 /user/hive/warehous
e/q1_orc
drwxr-xr-x   - hadoop supergroup          0 2020-03-02 21:13 /user/hive/warehous
e/q1_parquet
drwxr-xr-x   - hadoop supergroup          0 2020-03-02 21:22 /user/hive/warehous
e/q1_text
drwxr-xr-x   - hadoop supergroup          0 2020-02-20 21:45 /user/hive/warehous
e/salaries
bitnami@debian:~$
```

Display the sizes of the directories.

```
bitnami@debian:~$ hdfs dfs -du -h /user/hive/warehouse
215.6 K  215.6 K  /user/hive/warehouse/allstar
34       34       /user/hive/warehouse/departments
102.3 K  102.3 K  /user/hive/warehouse/fame
953.9 K  953.9 K  /user/hive/warehouse/people
51.4 M   51.4 M   /user/hive/warehouse/q1_avro
1.8 M    1.8 M    /user/hive/warehouse/q1_orc
3.1 M    3.1 M    /user/hive/warehouse/q1_parquet
49.7 M   49.7 M   /user/hive/warehouse/q1_text
0        0        /user/hive/warehouse/salaries
bitnami@debian:~$
```

The Avro format with 51.4M takes up most space, followed by the text format with 49.7M. The Parquet file format stores the data in a 3.1M file and the ORC format has the highest compression and stores the data in a 1.8M file.

Run the `describe formatted` commands for all four tables in Hive. The `describe formatted` command returns the detailed table information in a clean manner.

`describe formatted q1_avro;`

```
        COLUMN_STATS_ACCURATE   {\"BASIC_STATS\":\"true\",\"COLUMN_STATS\":{\"ar
r15\":\"true\",\"del15\":\"true\",\"dest\":\"true\",\"month\":\"true\",\"origin\
":\"true\",\"year\":\"true\"}}
        bucketing_version       2
        numFiles                1
        numRows                 1749235
        rawDataSize             0
        totalSize               53876609
        transient_lastDdlTime   1583184841

# Storage Information
SerDe Library:          org.apache.hadoop.hive.serde2.avro.AvroSerDe
InputFormat:            org.apache.hadoop.hive.ql.io.avro.AvroContainerInputForm
at
OutputFormat:           org.apache.hadoop.hive.ql.io.avro.AvroContainerOutputFor
mat
Compressed:             No
Num Buckets:            -1
Bucket Columns:         []
Sort Columns:           []
Storage Desc Params:
        field.delim             ,
        serialization.format    ,
Time taken: 2.01 seconds, Fetched: 37 row(s)
hive> _
```

`describe formatted q1_parquet;`

```
r15\":\"true\",\"del15\":\"true\",\"dest\":\"true\",\"month\":\"true\",\"origin\
":\"true\",\"year\":\"true\"}}
        bucketing_version       2
        numFiles                1
        numRows                 1749235
        rawDataSize             10495410
        totalSize               3302242
        transient_lastDdlTime   1583184799

# Storage Information
SerDe Library:          org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSe
rDe
InputFormat:            org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputF
ormat
OutputFormat:           org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutput
Format
Compressed:             No
Num Buckets:            -1
Bucket Columns:         []
Sort Columns:           []
Storage Desc Params:
        field.delim             ,
        serialization.format    ,
Time taken: 0.373 seconds, Fetched: 37 row(s)
hive> _
```

`describe formatted q1_orc;`

```
Table Type:             MANAGED_TABLE
Table Parameters:
        COLUMN_STATS_ACCURATE   {\"BASIC_STATS\":\"true\",\"COLUMN_STATS\":{\"ar
r15\":\"true\",\"del15\":\"true\",\"dest\":\"true\",\"month\":\"true\",\"origin\
":\"true\",\"year\":\"true\"}}
        bucketing_version       2
        numFiles                1
        numRows                 1749235
        rawDataSize             629724592
        totalSize               1935075
        transient_lastDdlTime   1583184749

# Storage Information
SerDe Library:          org.apache.hadoop.hive.ql.io.orc.OrcSerde
InputFormat:            org.apache.hadoop.hive.ql.io.orc.OrcInputFormat
OutputFormat:           org.apache.hadoop.hive.ql.io.orc.OrcOutputFormat
Compressed:             No
Num Buckets:            -1
Bucket Columns:         []
Sort Columns:           []
Storage Desc Params:
        field.delim             ,
        serialization.format    ,
Time taken: 0.42 seconds, Fetched: 37 row(s)
hive> _
```

```
describe formatted q1_text;
```

```
Retention:              0
Location:               hdfs://localhost:8020/user/hive/warehouse/q1_text
Table Type:             MANAGED_TABLE
Table Parameters:
        bucketing_version       2
        numFiles                1
        numRows                 0
        rawDataSize             0
        totalSize               52109240
        transient_lastDdlTime   1583184171

# Storage Information
SerDe Library:          org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe
InputFormat:            org.apache.hadoop.mapred.TextInputFormat
OutputFormat:           org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputForm
at
Compressed:             No
Num Buckets:            -1
Bucket Columns:         []
Sort Columns:           []
Storage Desc Params:
        field.delim             ,
        serialization.format    ,
Time taken: 0.293 seconds, Fetched: 36 row(s)
hive> _
```