**Aviation Data Analysis Using Apache Pig**

There are 2 different data-sets, Airports.csv and Delayed Flights.csv and Piggybankjar copy them to Airline folder under /home/acadgild.

**Problem Statement 1**

Find out the top 5 most visited destinations.

**1.** REGISTER '/home/acadgild/Airline/piggybank.jar';

We are registering the piggybank jar in order to use the CSVExcelStorage class.

**2**. A = load '/home/acadgild/Airline/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO\_MULTILINE','UNIX','SKIP\_INPUT\_HEADER');

In relation A, we are loading the dataset using CSVExcelStorage because of its effective technique to handle double quotes and headers.

1. B = foreach A generate (int)$1 as year, (int)$10 as flight\_num, (chararray)$17 as origin,(chararray) $18 as dest;

In relation B, we are generating the columns that are required for processing and explicitly typecasting each of them.

1. C = filter B by dest is not null;

In relation C, we are filtering the null values from the “dest” column.

1. D = group C by dest;

In relation D, we are grouping relation C by “dest.”

1. E = foreach D generate group, COUNT(C.dest);

In relation E, we are generating the grouped column and the count of each.

1. F = order E by $1 DESC;

Relation F and Result is used to order and limit the result to top 5.

1. Result = LIMIT F 5;

These are the steps to find the top 5 most visited destinations. However, adding few more steps in this process, we will be using another table to find the city name and country as well.

1. A1 = load '/home/acadgild/Airline/airports.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO\_MULTILINE','UNIX','SKIP\_INPUT\_HEADER');

In relation A1, we are loading another table to which we will look-up and find the city as well as the country.

1. A2 = foreach A1 generate (chararray)$0 as dest, (chararray)$2 as city, (chararray)$4 as country;

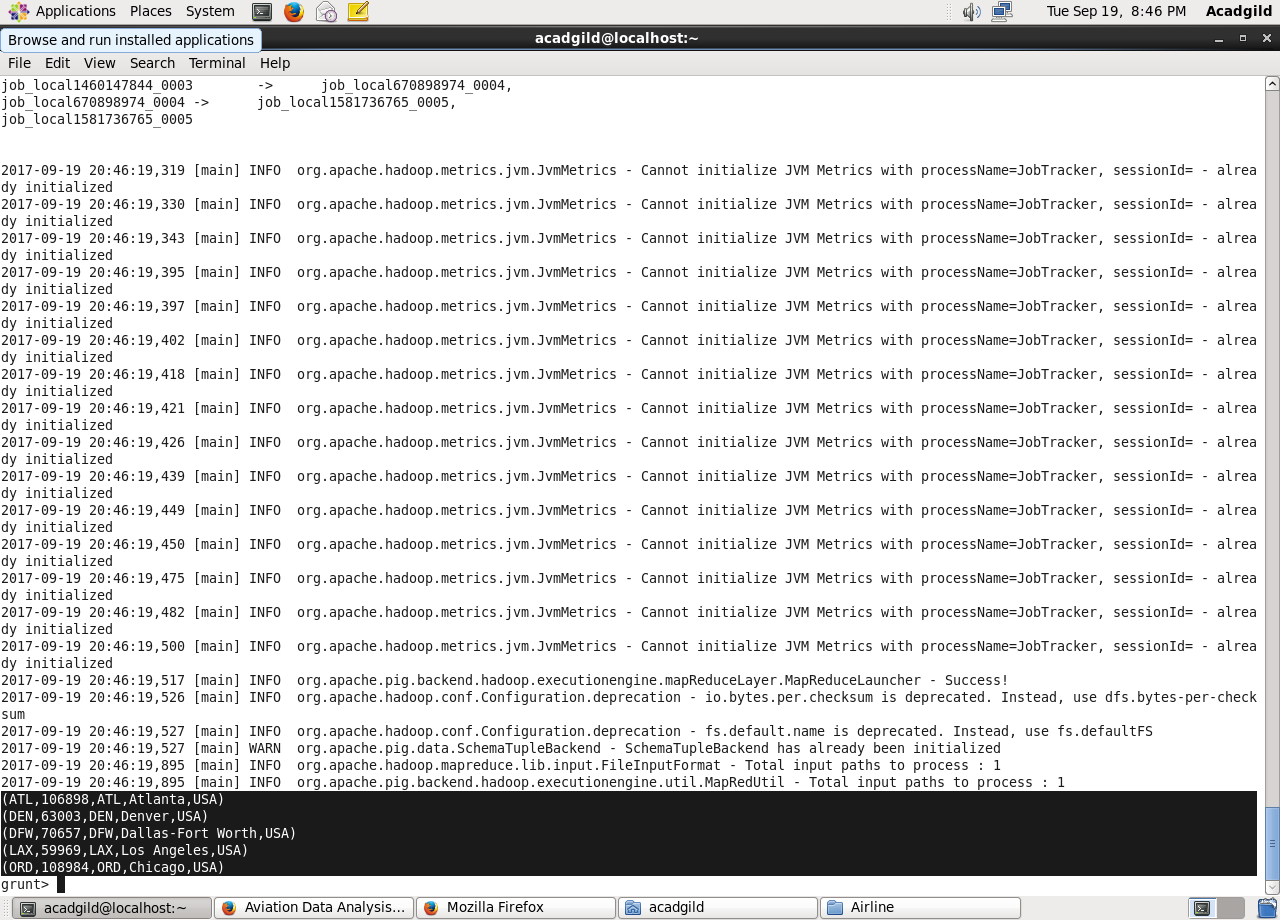
In relation A2, we are generating dest, city, and country from the previous relation.

1. joined\_table = join Result by $0, A2 by dest;

In relation joined\_table, we are joining Result and A2 based on a common column, i.e., “dest”

1. dump joined\_table;

Finally, using dump, we are printing the result.



**Problem Statement 2**

Which month has seen the most number of cancellations due to bad weather?

1. REGISTER '/home/acadgild/Airline/piggybank.jar';

We are registering piggybank jar in order to use the CSVExcelStorage class.

1. A = load '/home/acadgild/Airline/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO\_MULTILINE','UNIX','SKIP\_INPUT\_HEADER');

In relation A, we are loading the dataset using CSVExcelStorage because of its effective technique to handle double quotes and header.

3. B = foreach A generate (int)$2 as month,(int)$10 as flight\_num,(int)$22 as cancelled,(chararray)$23 as cancel\_code;

In relation B, we are generating the columns which are required for processing and explicitly typecasting each of them.

4. C = filter B by cancelled == 1 AND cancel\_code =='B';

In relation C, we are filtering the data based on cancellation and cancellation code, i.e., canceled = 1 means flight have been canceled and cancel\_code = ‘B’ means the reason for cancellation is “weather.” So relation C will point to the data which consists of canceled flights due to bad weather.

5. D = group C by month;

In relation D, we are grouping the relation C based on every month.

6. E = foreach D generate group, COUNT(C.cancelled);

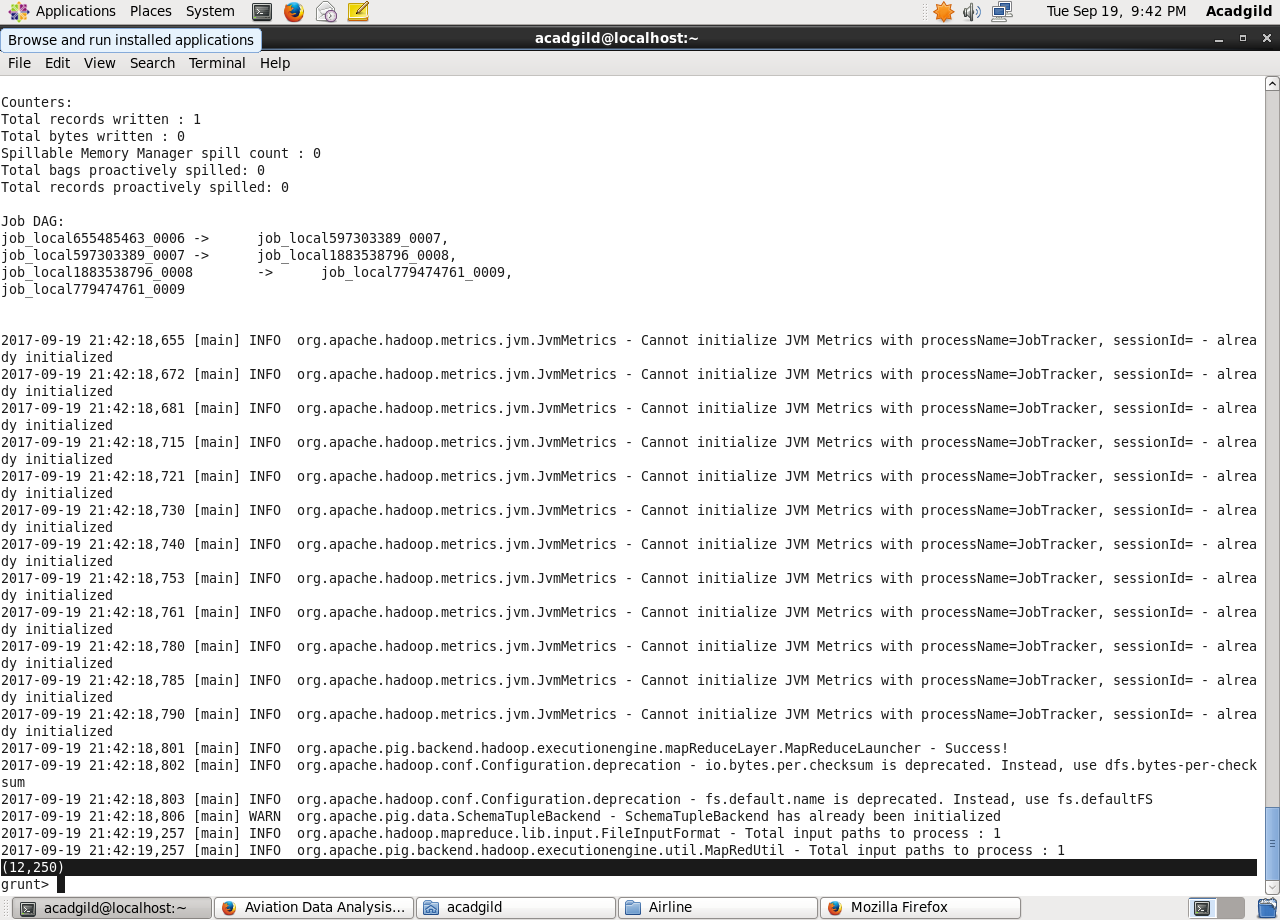
In relation E, we are finding the count of canceled flights every month.

7. F= order E by $1 DESC;

Result = limit F 1;

Relation F and Result is for ordering and finding the top month based on cancellation

1. dump Result;



**Problem Statement 3**

Top ten origins with the highest AVG departure delay

1. REGISTER '/home/acadgild/Airline/piggybank.jar';

2. A = load '/home/acadgild/Airline/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO\_MULTILINE','UNIX','SKIP\_INPUT\_HEADER');

3. B1 = foreach A generate (int)$16 as dep\_delay, (chararray)$17 as origin;

1. C1 = filter B1 by (dep\_delay is not null) AND (origin is not null);

In relation C1, we are removing the null values fields present if any.

1. D1 = group C1 by origin;

In relation D1, we are grouping the data based on column “origin.”

1. E1 = foreach D1 generate group, AVG(C1.dep\_delay);

In relation E1, we are finding average delay from each unique origin.

Relations named Result and Top\_ten are ordering the results in descending order and printing the top ten values.

7. Result = order E1 by $1 DESC;

8. Top\_ten = limit Result 10;

However, rather than generating just the code of origin, we will be following a few more steps to find some more details like country and city.

1. Lookup = load '/home/acadgild/Airline/airports.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO\_MULTILINE','UNIX','SKIP\_INPUT\_HEADER');

In the relation Lookup, we are loading another table to which we will look up and find the city as well as the country.

10. Lookup1 = foreach Lookup generate (chararray)$0 as origin, (chararray)$2 as city, (chararray)$4 as country;

In the relation Lookup1, we are generating the destination, city, and country from the previous relation.

11.Joined = join Lookup1 by origin, Top\_ten by $0;

In the relation Joined, we are joining relation Top\_ten and Lookup1 based on common a column, i.e., “origin.”

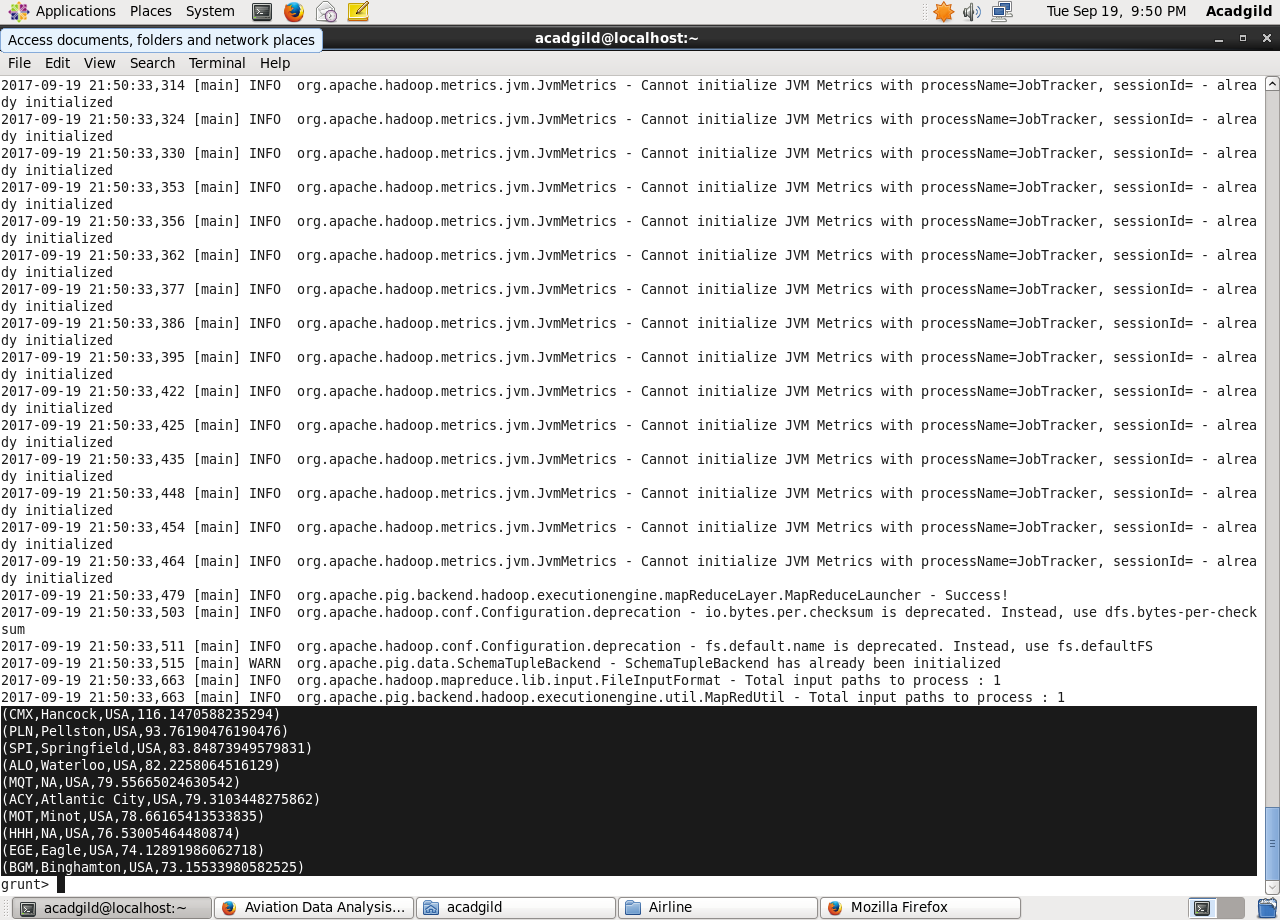
12 . Final = foreach Joined generate $0,$1,$2,$4;

13. Final\_Result = ORDER Final by $3 DESC;

1. dump Final\_Result;

In the relation Final, we are generating required columns from the Joined table.

Finally, we are ordering and printing the results.



**Problem Statement 4**

Which route (origin & destination) has seen the maximum diversion?

1. REGISTER '/home/acadgild/Airline/piggybank.jar';

We are registering piggybank jar in order to use CSVExcelStorage class.

2. A = load '/home/acadgild/Airline/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO\_MULTILINE','UNIX','SKIP\_INPUT\_HEADER');

In relation A, we are loading the dataset using CSVExcelStorage because of its effective technique to handle double quotes and headers.

3. B = FOREACH A GENERATE (chararray)$17 as origin, (chararray)$18 as dest, (int)$24 as diversion;

In relation B, we are generating the columns which are required for processing and explicitly type-casting each of them.

4. C = FILTER B BY (origin is not null) AND (dest is not null) AND (diversion == 1);

In relation C, we are filtering the data based on “not null” and diversion =1. This will remove the null records, if any, and give the data corresponding to the diversion taken.

5. D = GROUP C by (origin,dest);

In relation D, we are grouping the data based on origin and destination.

6. E = FOREACH D generate group, COUNT(C.diversion);

Relation E finds the count of diversion taken per unique origin and destination.

7. F = ORDER E BY $1 DESC;

Result = limit F 10;

Relations F and Result orders the result and produces top 10 results.

dump Result;

