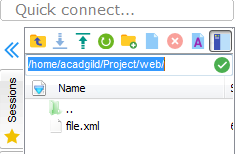
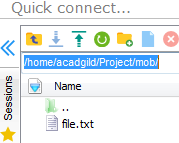
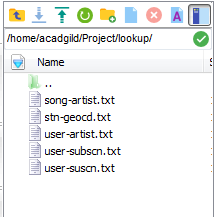
**Data Analysis (IMPLEMETED IN SPARK)**

It is not only the data which is important, rather it is the insight it can be used to generate

important. Once we have made the data ready for analysis, we have to perform below analysis on a

daily basis.

files are downloaded in respective folder as given in below screen shot. 

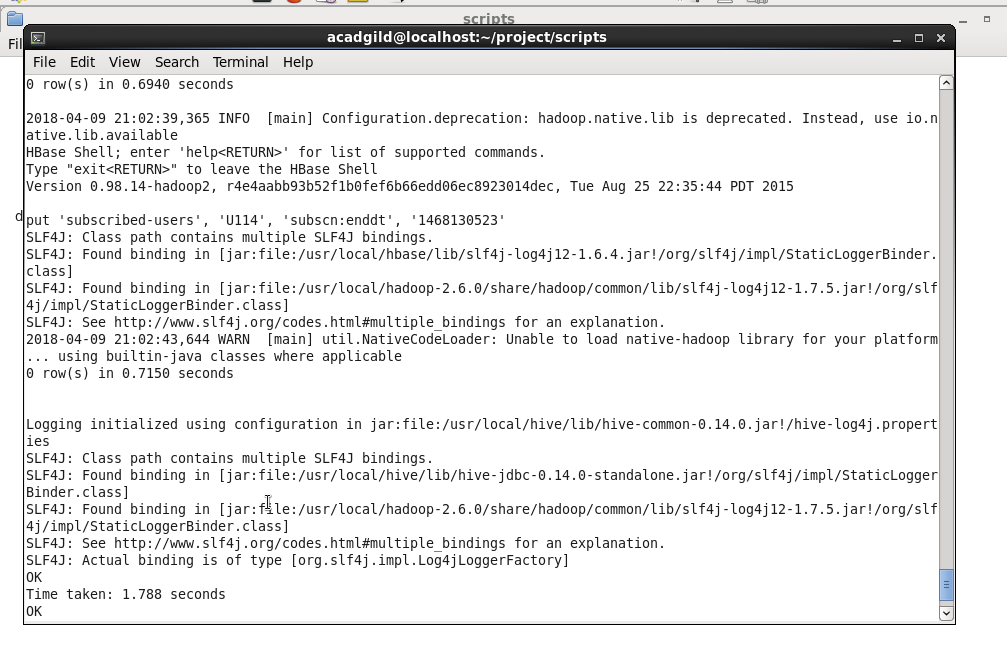
**Lookup Tables creation in HBASE**

By using the populate-lookup.sh script we will create lookup tables in Hbase. These tables have to be used in data formatting, data enrichment and analysis stage.

Below is scripts:



Below is screen shot after script execution:



Below tables are created using above script in hbase

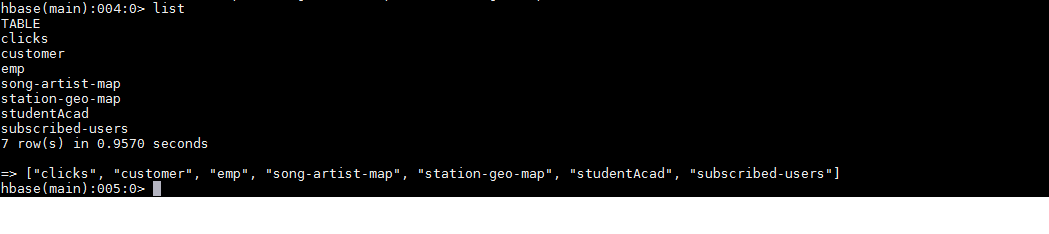
And table created in hive as shown in below screen shot

station-geo-map

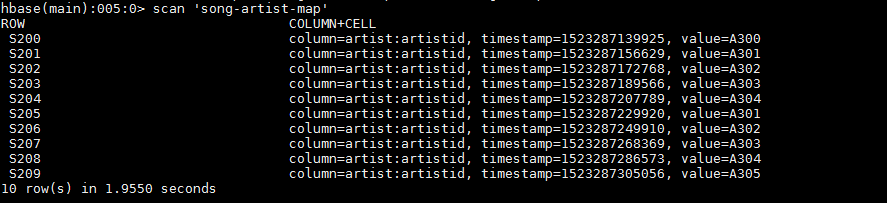
subscribed-users

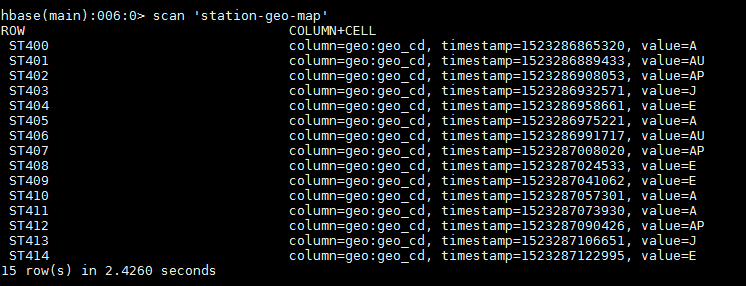
song-artist-map

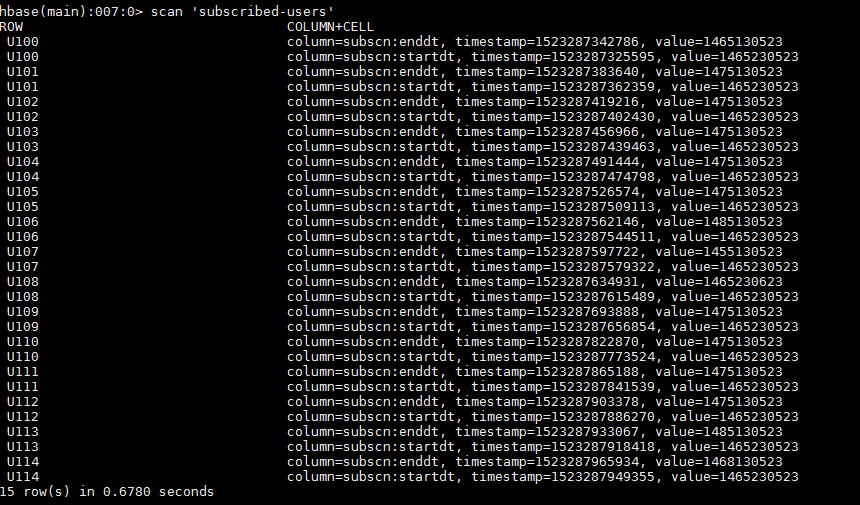
Now we can see the lookup tables in Hbase shell terminal as shown below:



Scan of all tables







Now Lookup table creation is completed. So now we need to link theses lookup tables in hive using the Hbase Storage Handler.

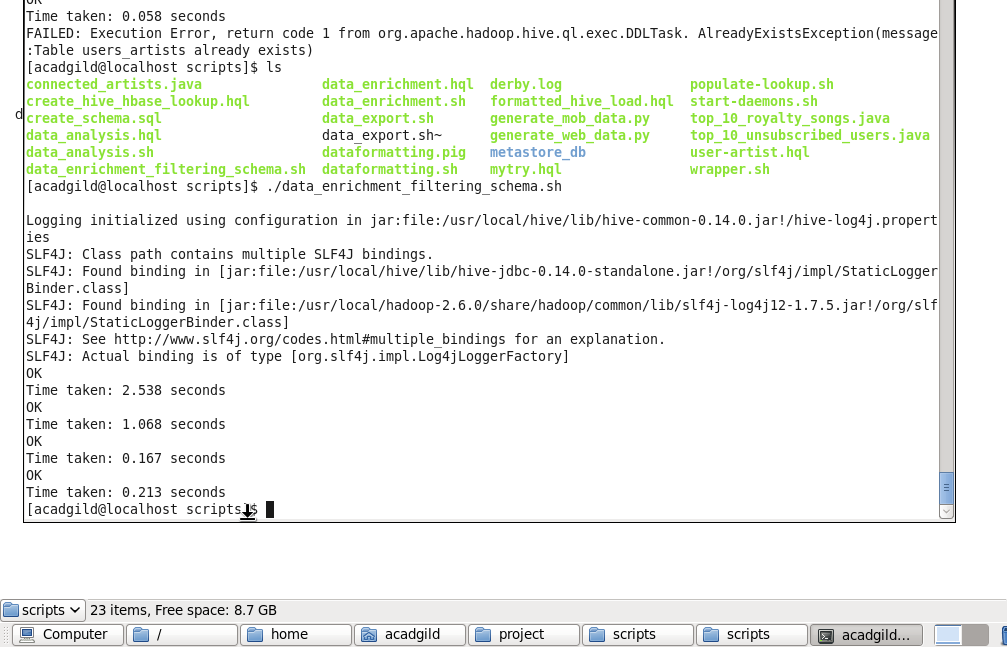
**Creating Hive Tables on the top of Hbase:**

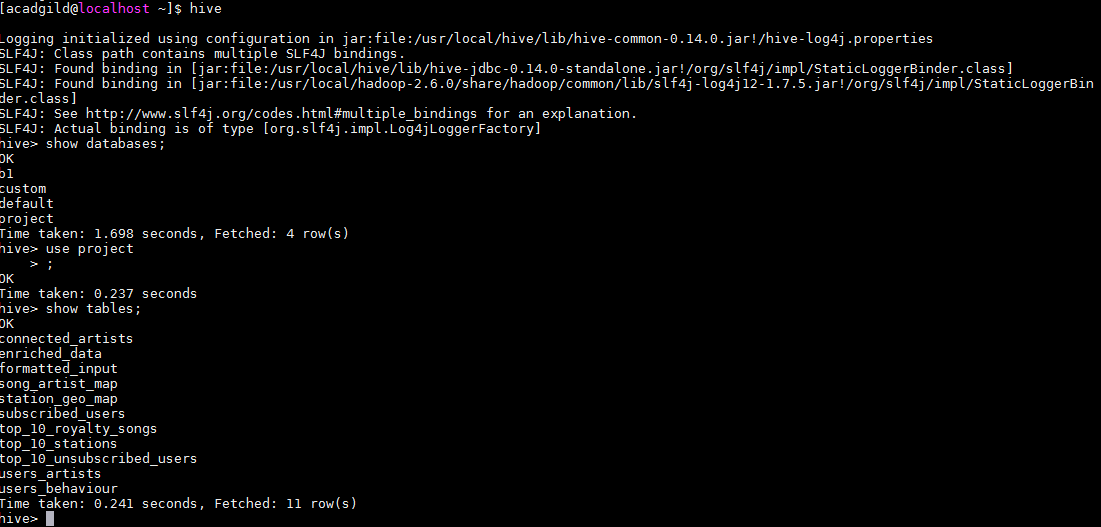
Using below scripts





Output after scripts execution





**Data Formatting:**

In this stage we are merging the data coming from both web applications and mobile applications and create a common table for analyzing purpose and create partitioned data based on batchid,

Running two scripts to format the data which is calling in scripts dataformatting .sh. They are





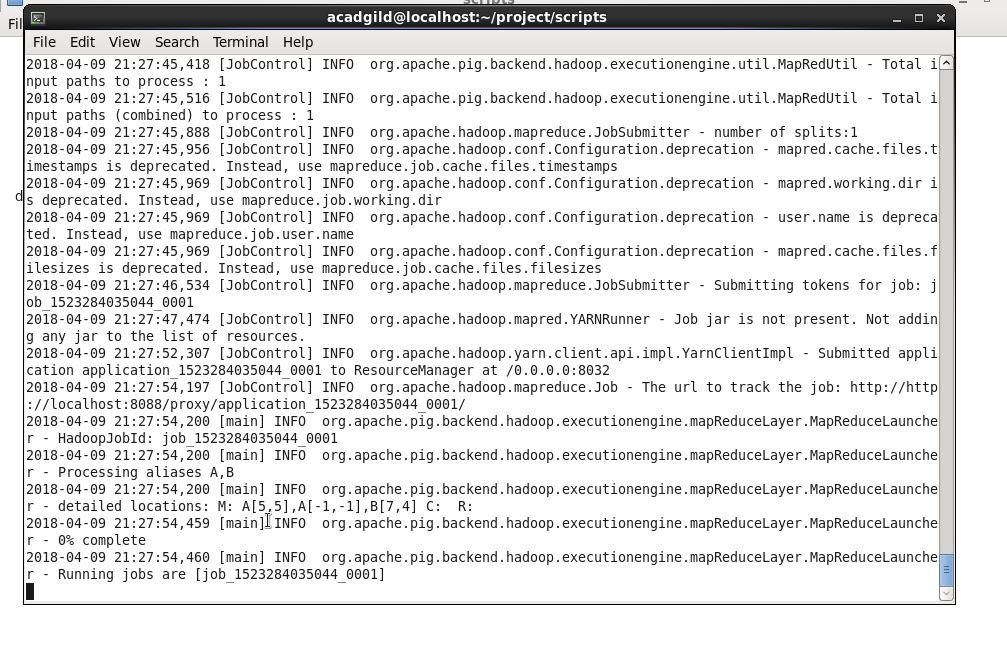


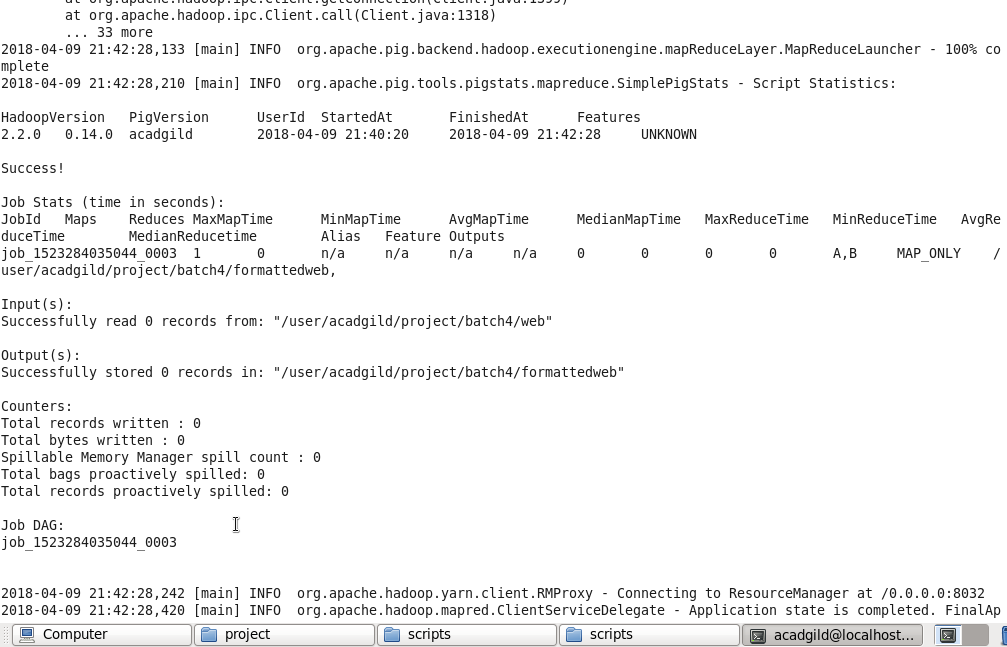
Dataformatting.sh calling two scripts formatted\_hive\_load.hql and dataformatting.pig.

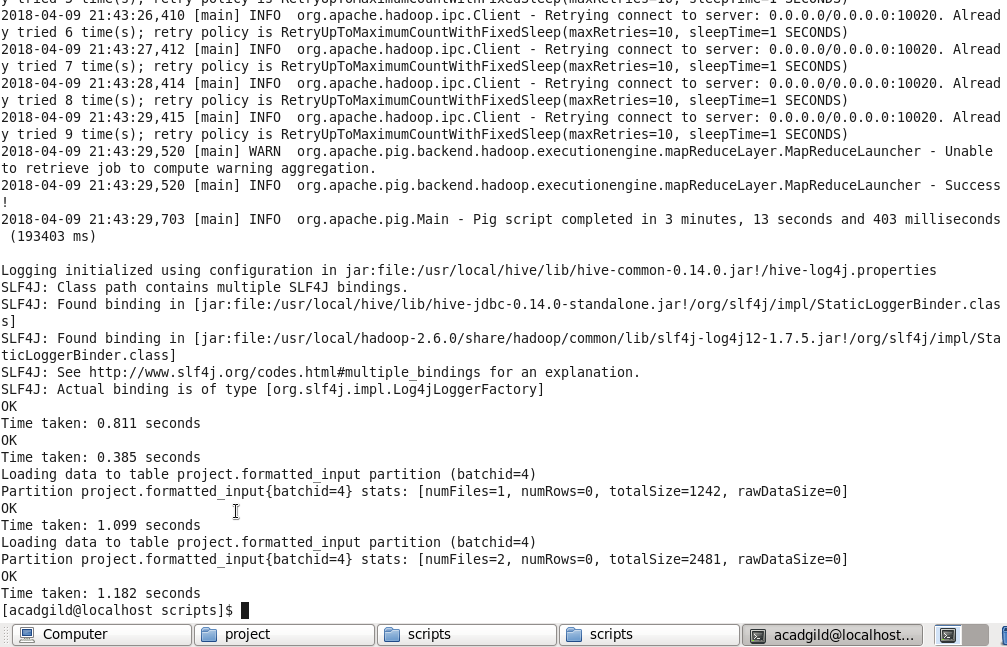
Pig script to parse the data from coming from web\_data.xml to csv format and partition both web and mob data based on based on batch ID’s

Hive script to load the parsed data from pig to hive and do analysis based on this data

Output screen of above scripts.

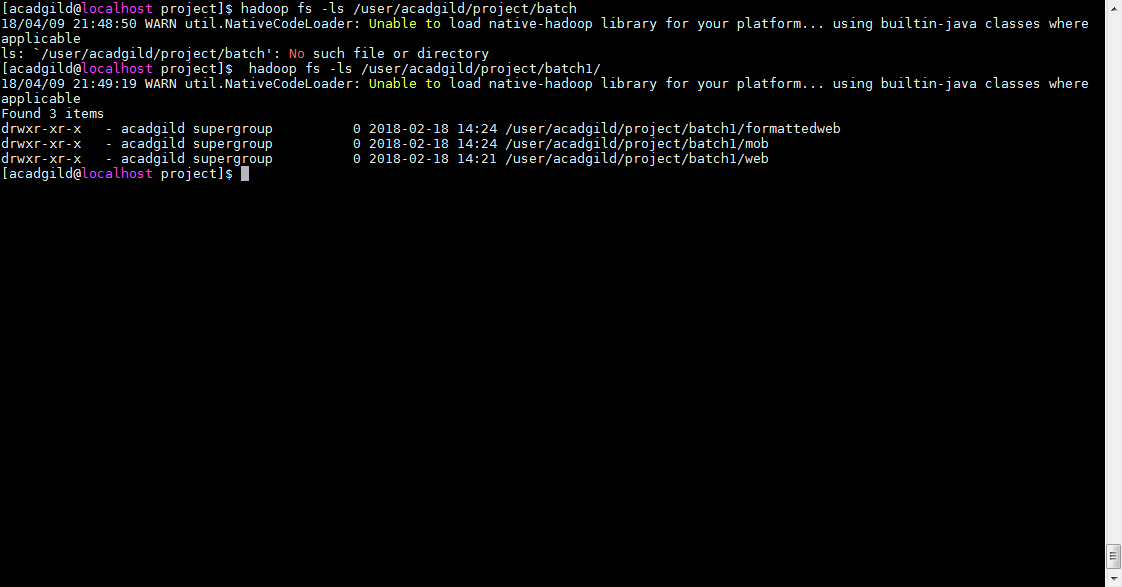




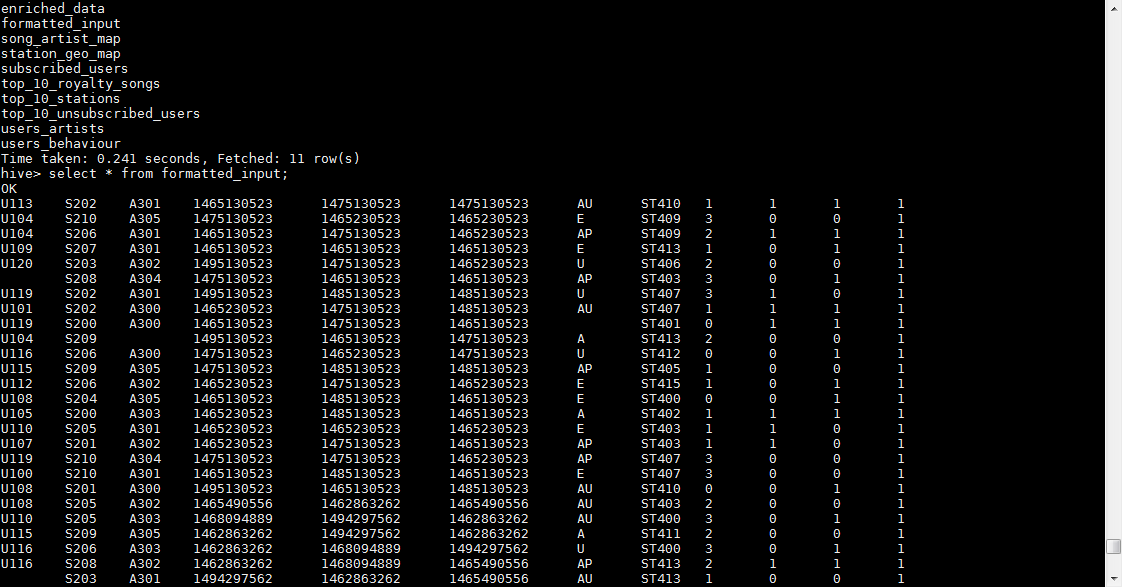


In the above screenshot we can see the dataformatting.pig along with the formatted\_hive\_load.hql executed successfully.

The output of dataformatting.sh script in HDFS folders:



DataFormatting.sh output in hive terminal



**Data Enrichment:**

In this phase we will enrich the data coming from web and mobile applications using the lookup table stored in Hbase and divide the records based on the enrichment rules into ‘pass’ and ‘fail’ records.

Rules for data enrichment

1. If any of like or dislike is NULL or absent, consider it as 0.

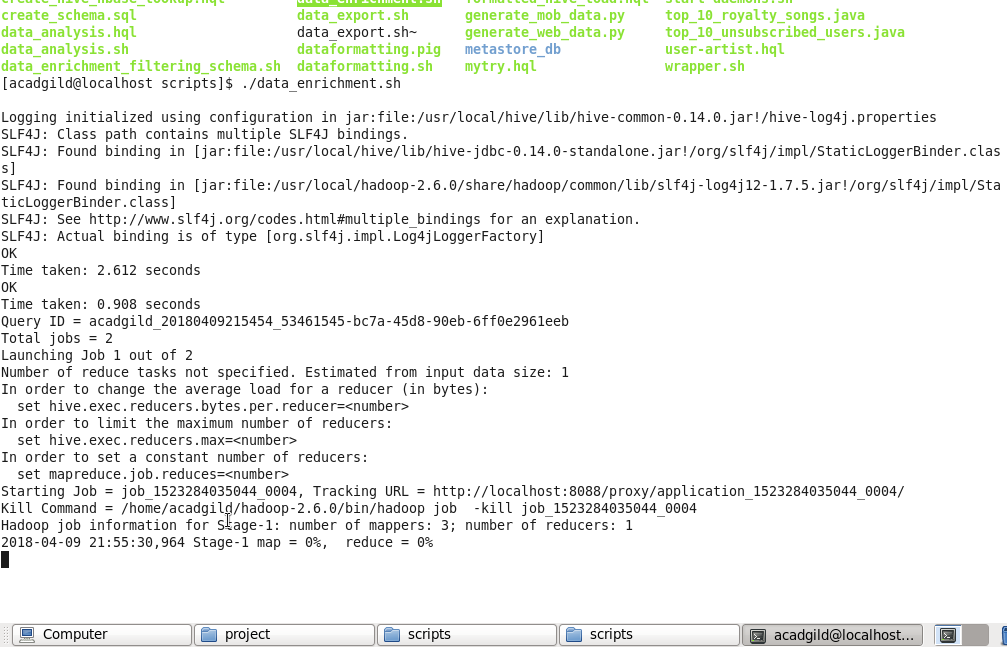
2. If fields like Geo\_cd and Artist\_id are NULL or absent, consult the lookup tables for fields Station\_id and Song\_id respectively to get the values of Geo\_cd and Artist\_id.

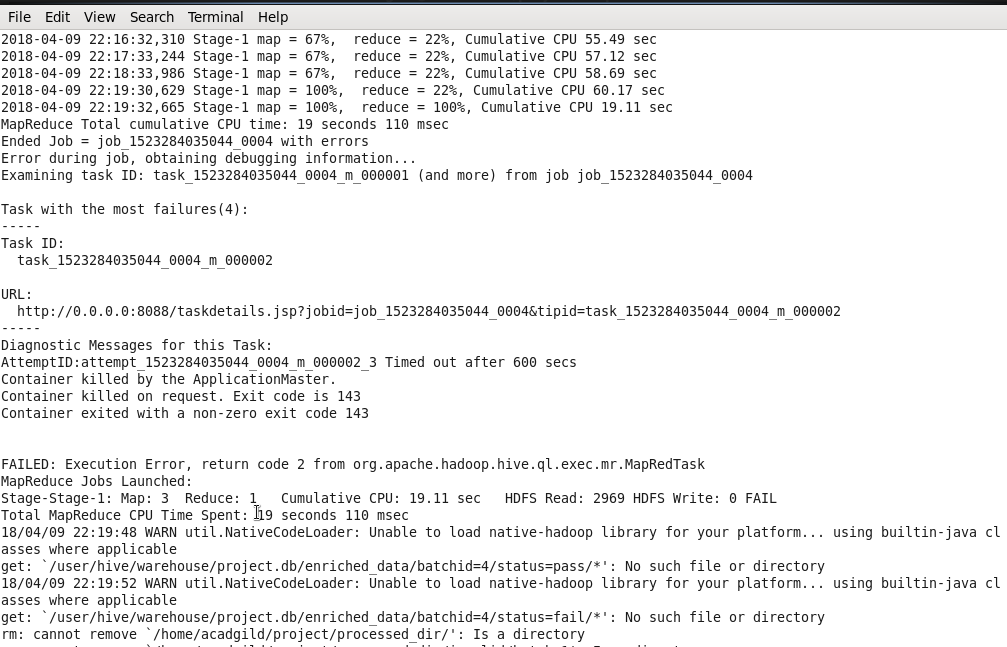
3. If corresponding lookup entry is not found, consider that record to be invalid

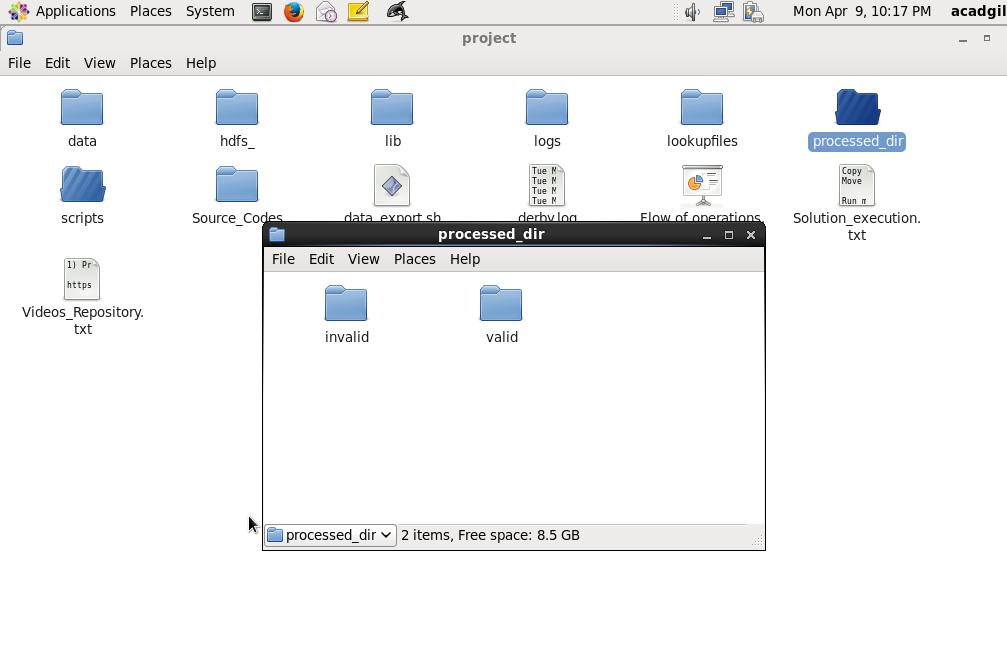
So based on the enrichment rules we will fill the null geo\_cd and artist\_id values with the help of corresponding lookup values in song-artist-map and station-geo-map tables in Hive-Hbase tables.

data\_enrichment.sh used for this ruls

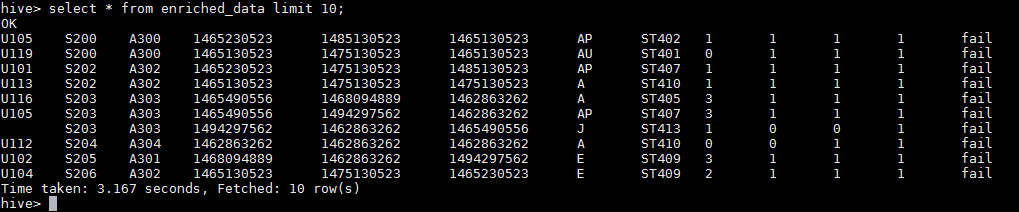


Result screen shot



At the end script will automatically divide the records based on status pass & fail and dump the result into processed\_dir folder with valid and invalid folders

In the below screenshot we have data for data enrichment table where we filled the null values of artist\_id and geo\_cd of formatted input with the help of lookup tables



Enrichment phase is executed successfully by applying all the rules of enrichment

**Data Analysis using spark:**

**enriched\_data and subscribed\_users data from hive table has been exported into txt file and then uploaded to spark VM for further analysis using spark**

Problem 1 : Determine top 10 station\_id(s) where maximum of songs were played,which were linked by unique users



Result screen shot

