**Assignment - 4**

[**DW/OLAP & ETL Exercises Using IBM Cloud**]

Rahul Midha Niravsinh Jadeja

B00766975 B00789139

1. **Application & Datasets**

**Application Name: Organization Expense System**

This application helps us in fetching department expenses and position expenses in an organization. It reduces manual computation of the results from the picture and thus saves a lot of time. In this application, we have made use of IBM DB2 Warehouse *[1]* to fetch the datasets. The GOSALESHR dataset has been used in the application. By applying various OLAP *[2]* features, we have been able to successfully fetch the following information from the datasets:

1. Total Expense of each Organization
2. Total Expense of each position in the organization

In this way, we can fetch the overall expense of each organization and the expense of each position in the organization using the application.

1. **Data Preparation ETL**

ETL *[3]* is a process in data warehousing which refers to extract, transform and load.

* 1. **Extract:** We have used schema from IBM DB2 warehouse named ‘GOSALESHR’, in which we have selected several tables based on our application which are as follows:

1. Employee Expense Details
2. Expense Type
3. Employee History
4. Organization
5. Position Summary
6. Position Lookup
7. Employee
8. Gender Lookup
   1. **Transformation:** For the implementation of this stage we have used Data Preparation Module from Google Cloud *[4]*. We have removed unnecessary columns from all the tables before uploading to the Google Cloud as this process becomes time consuming in Google Cloud, but it can also the done by editing the recipe and deleting the unnecessary columns. After uploading the data, we further checked for the valid values and removed the missing values from it.

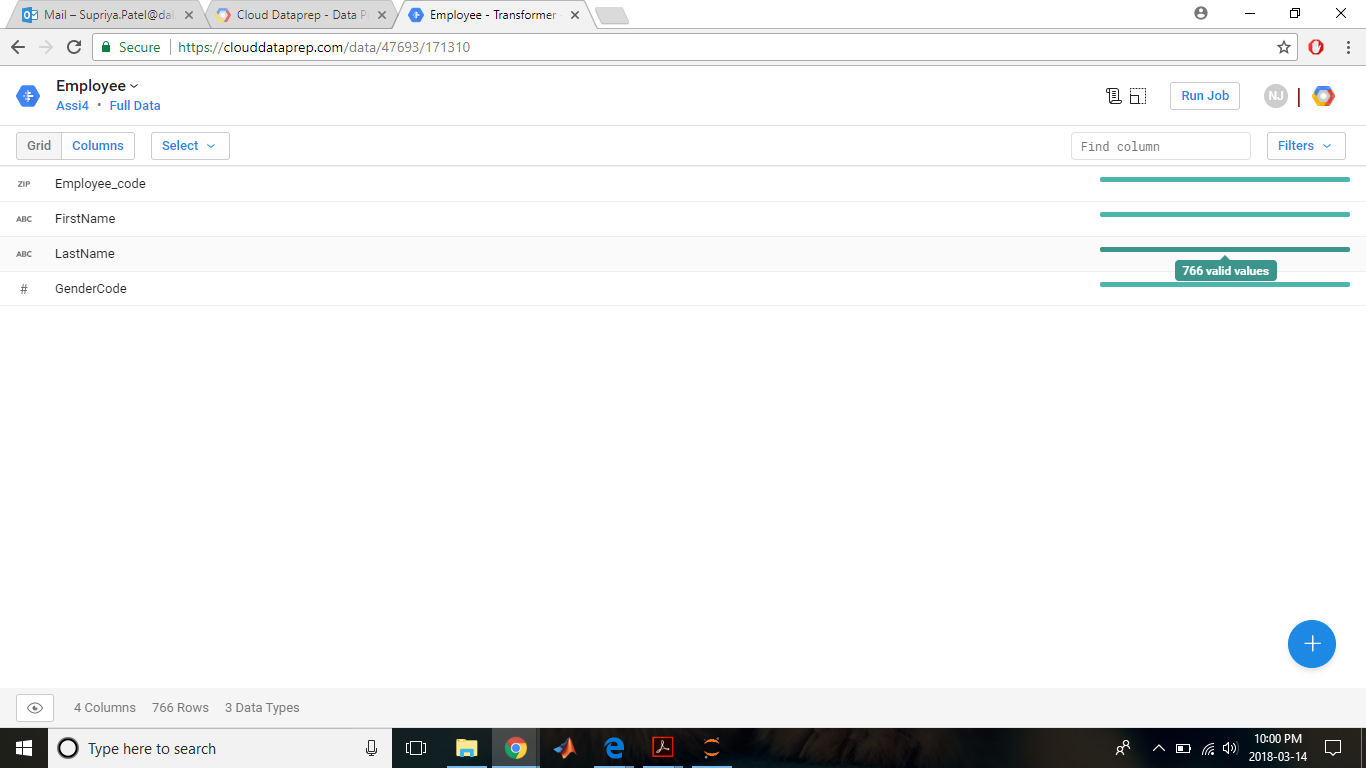


Figure 1: uploaded tables on Data Preparation module, Google Cloud

* 1. **Load:** For loading the data, we have utilized DB2 warehouse on cloud module from IBM Cloud *[5]*.

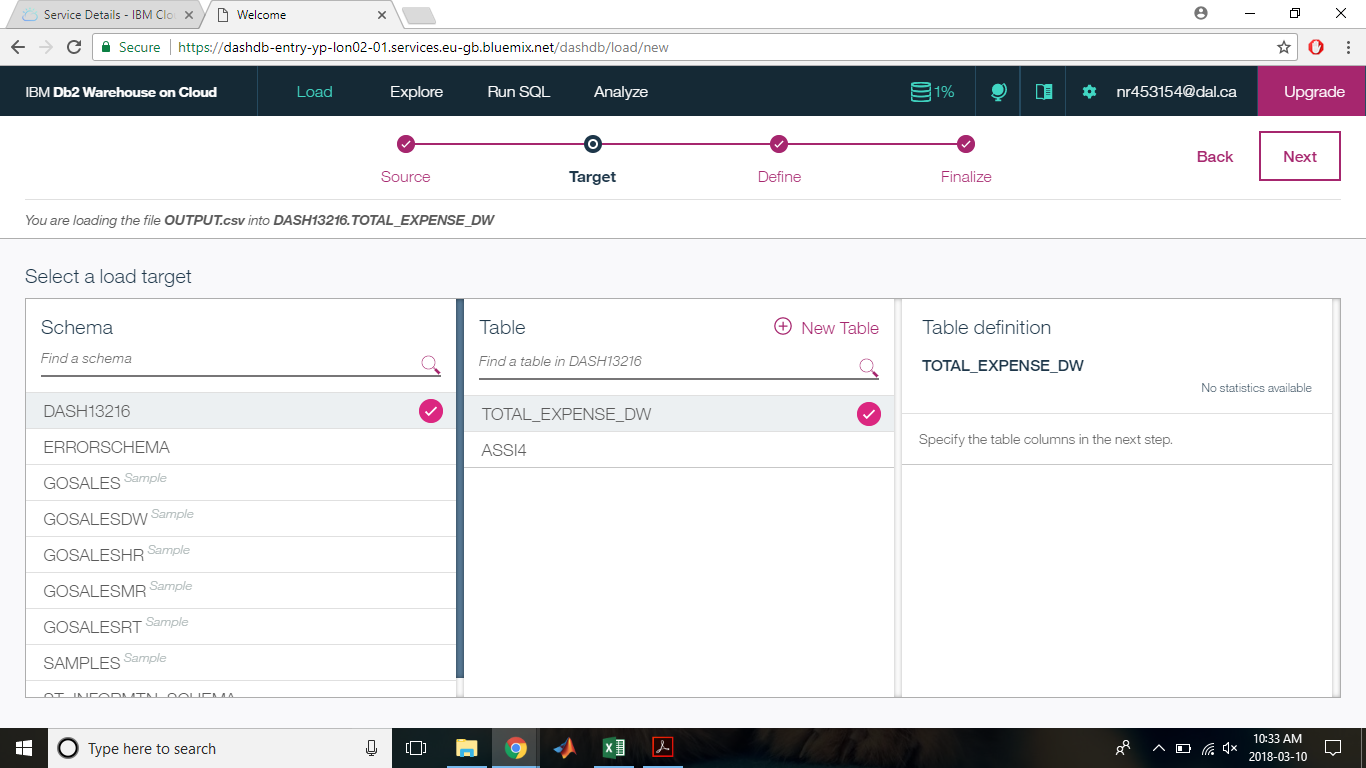


Figure 2: Loading the prepared dataset on IBM DB2 warehouse

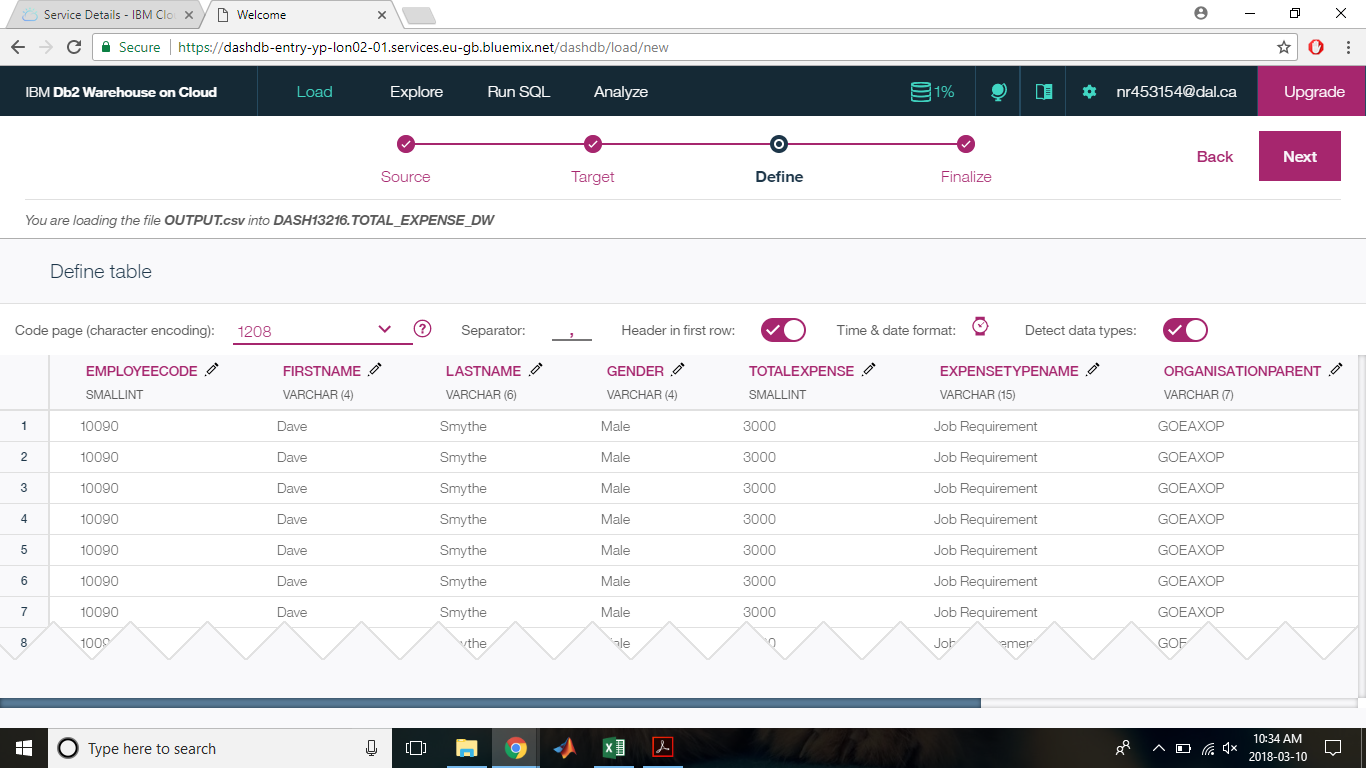


Figure 3: dataset uploading wizard on IBM DB2 warehouse – change type of data

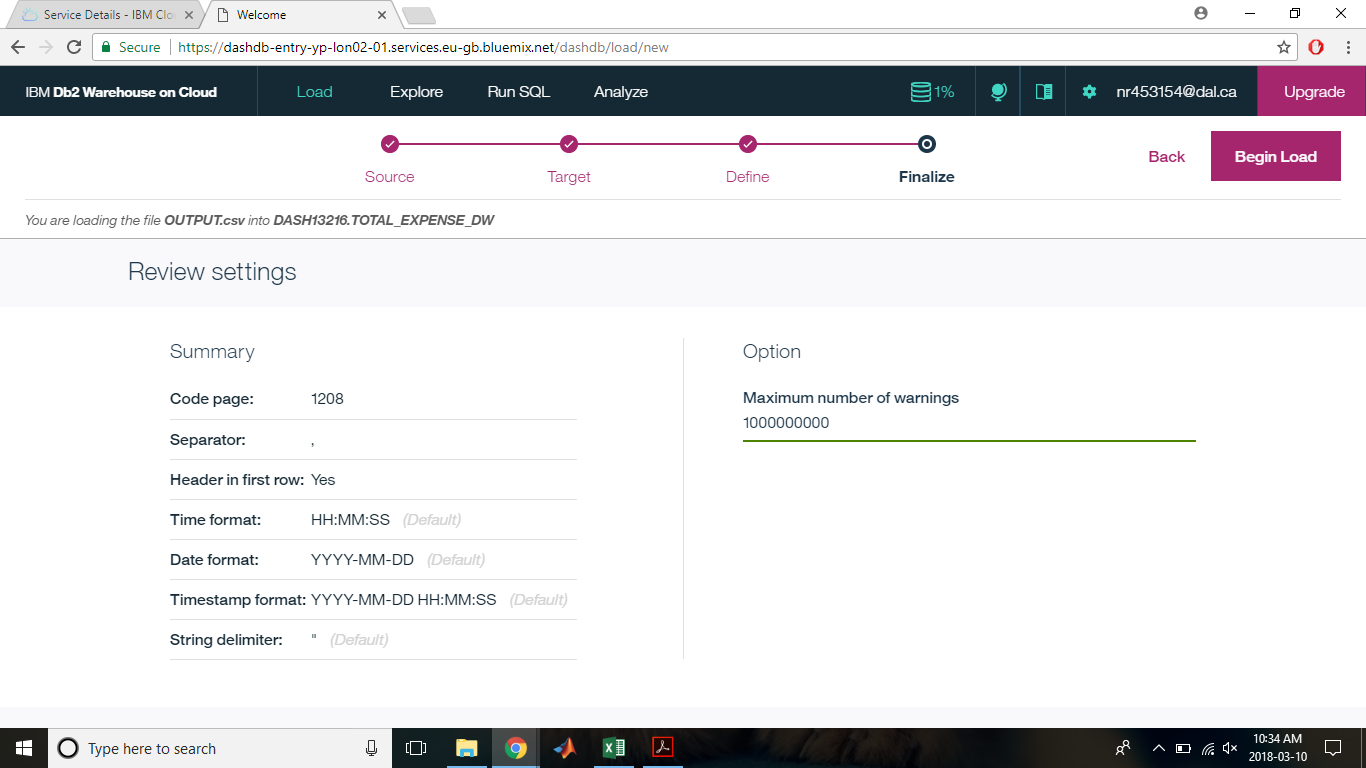


Figure 4: dataset uploading wizard on IBM DB2 warehouse – reviewing setting

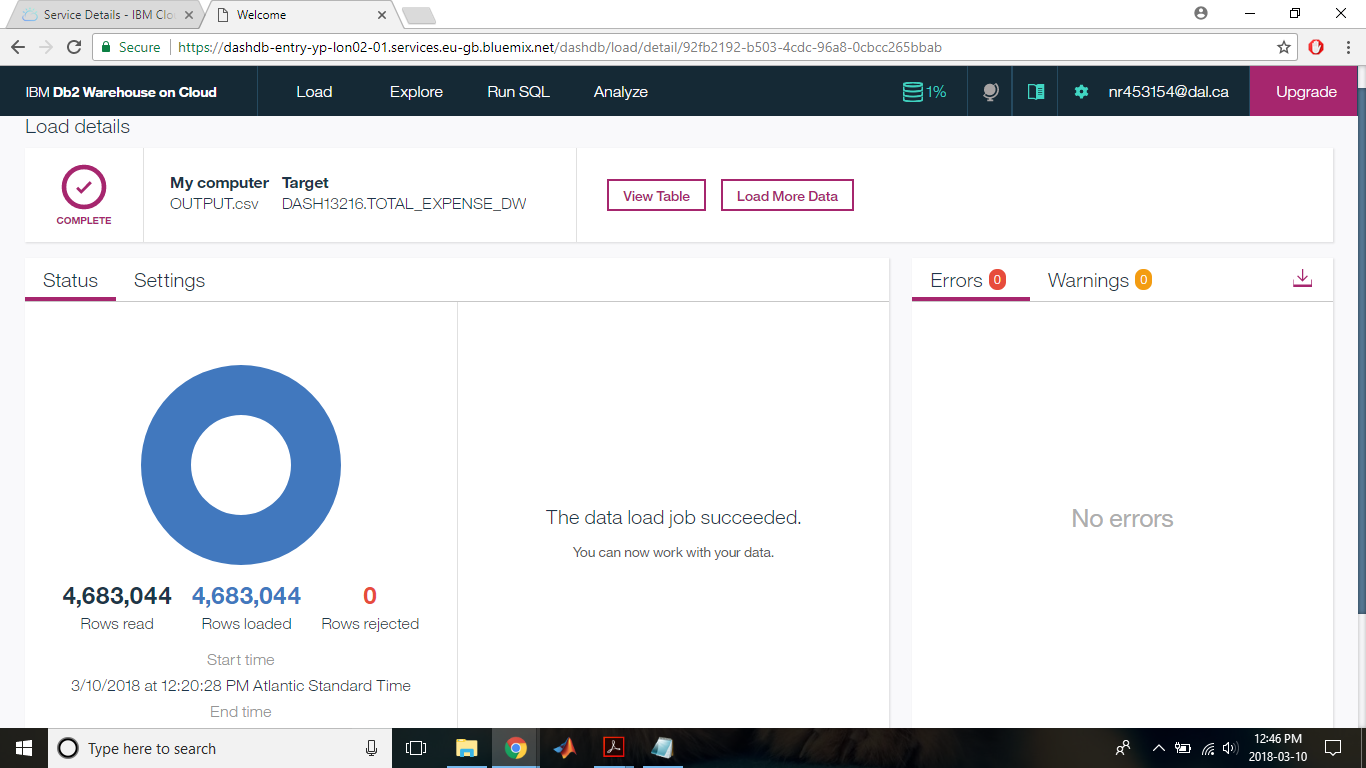


Figure 5: dataset uploading wizard on IBM DB2 warehouse – Confirmation of uploaded data

1. **DW Design & Implementation**

The implementation in Data Preparation module has been done by joining the edited columns and finding its output by clicking on run job.

From the following data it can be seen how the recipes have been joined and the resultant dataset and dataflow we obtain.

**3.1 Tables Joining Operation**

We have connected Employee and Gender Look up table using the Gender Code.

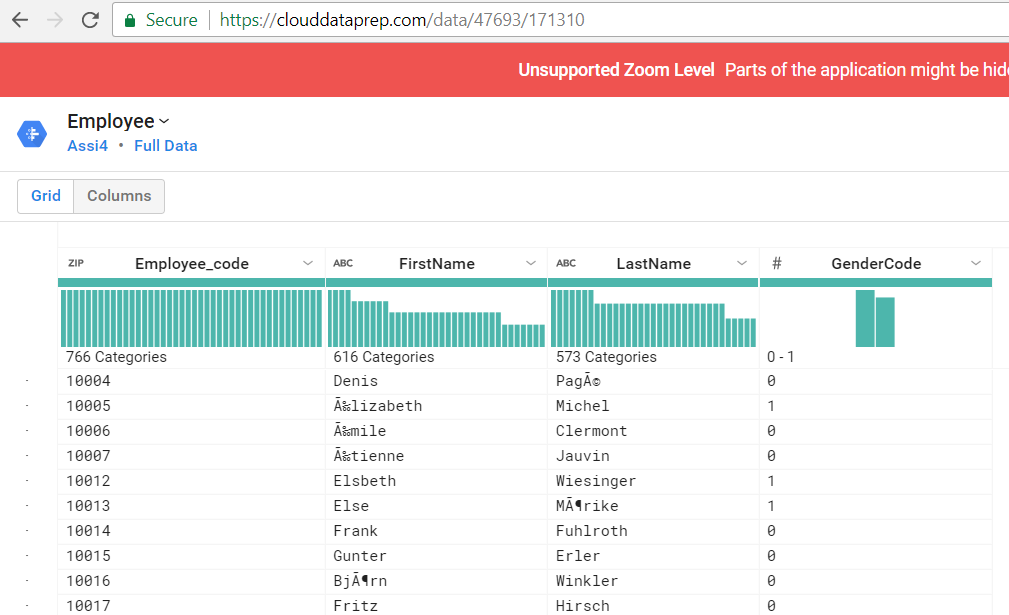
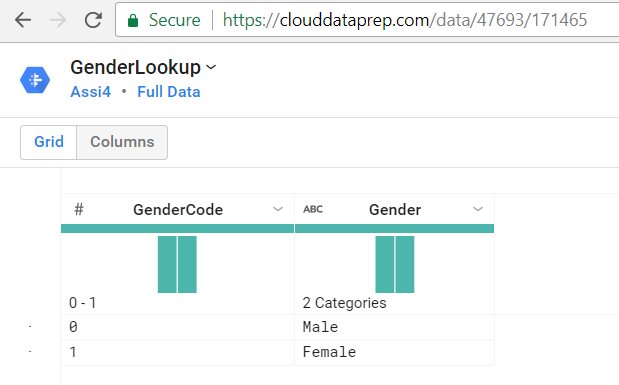
****

Figure 6: Joining of 2 Tables in Data Preparation Module, Google Could – Employee and Gender Look Up table

Organization and Employee History have been connected using the Organization Code

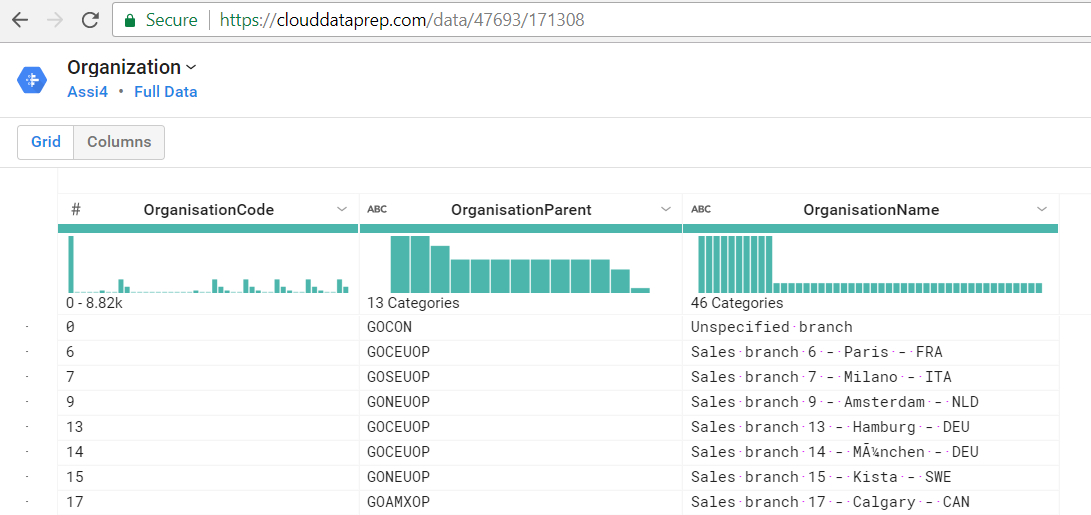
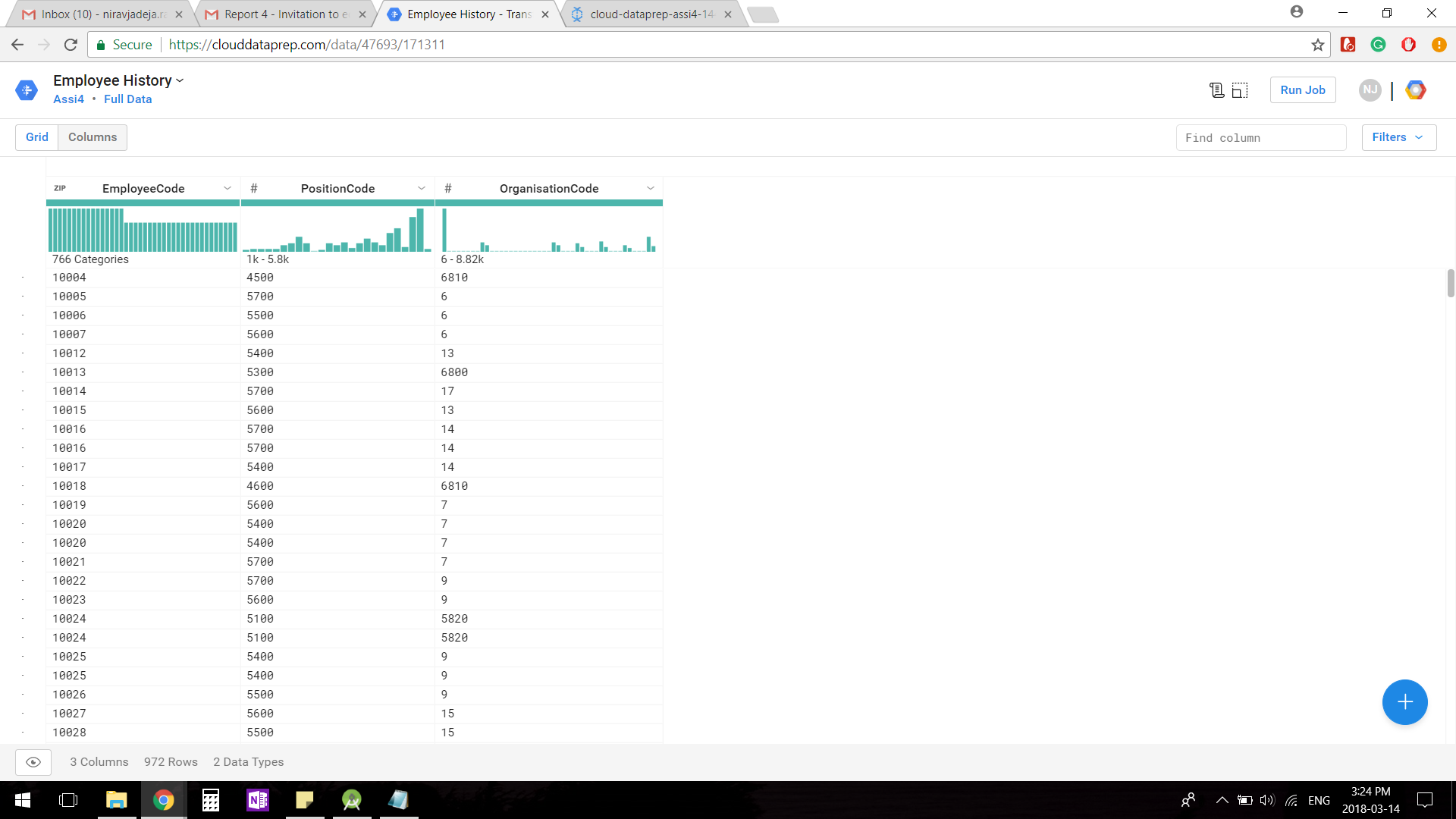


Figure 7: Joining of 2 Tables in Data Preparation Module, Google Could – Organization and Employee History

Employee Expense Detail and Expense Type have been connected using Expense Type Code

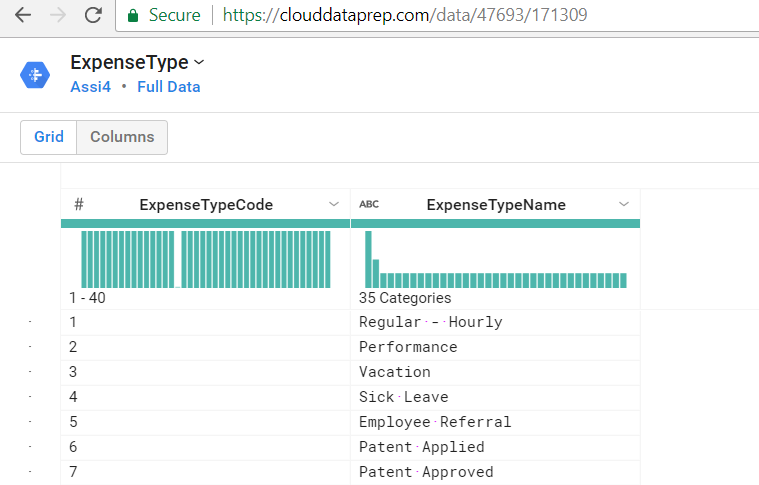
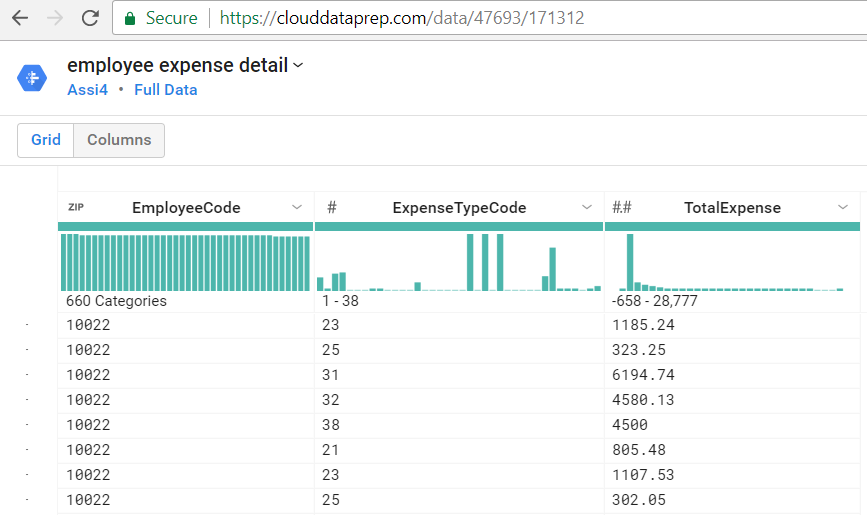


Figure 8: Joining of 2 Tables in Data Preparation Module, Google Could – Employee Expense Detail and Expense Type

Position Summary and Position\_lookup have been connected using Position Code

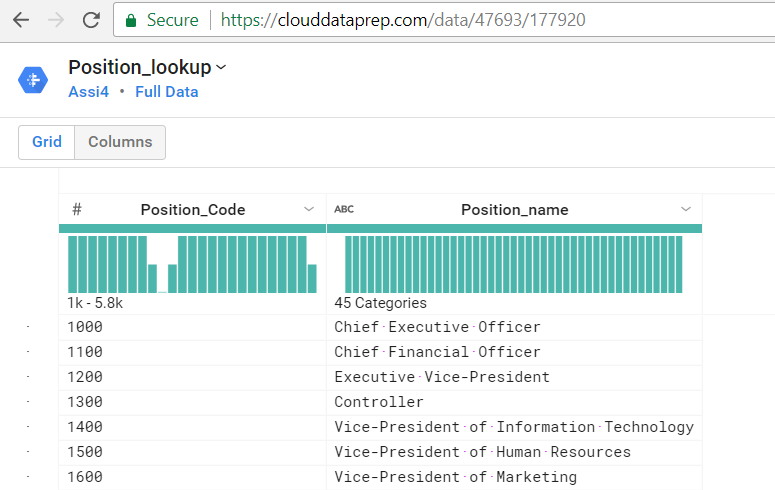
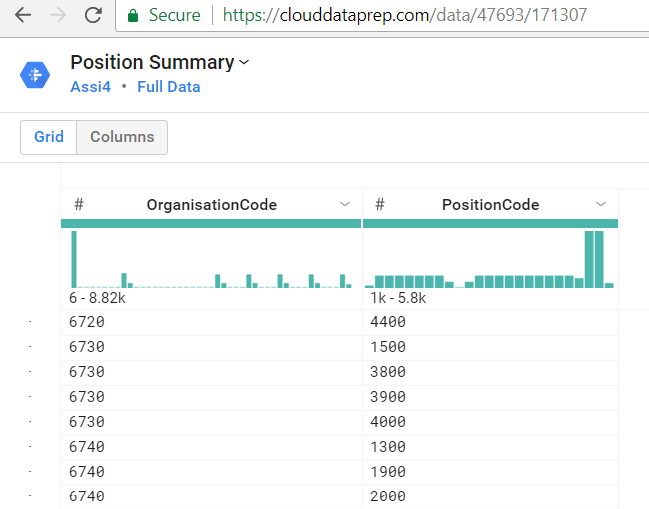


Figure 9: Joining of 2 Tables in Data Preparation Module, Google Could – Position Summary and Position Code

* 1. **Data Flow Diagram**

Here, we have attached the data flow diagram of final dataset which was processed on Google Cloud.

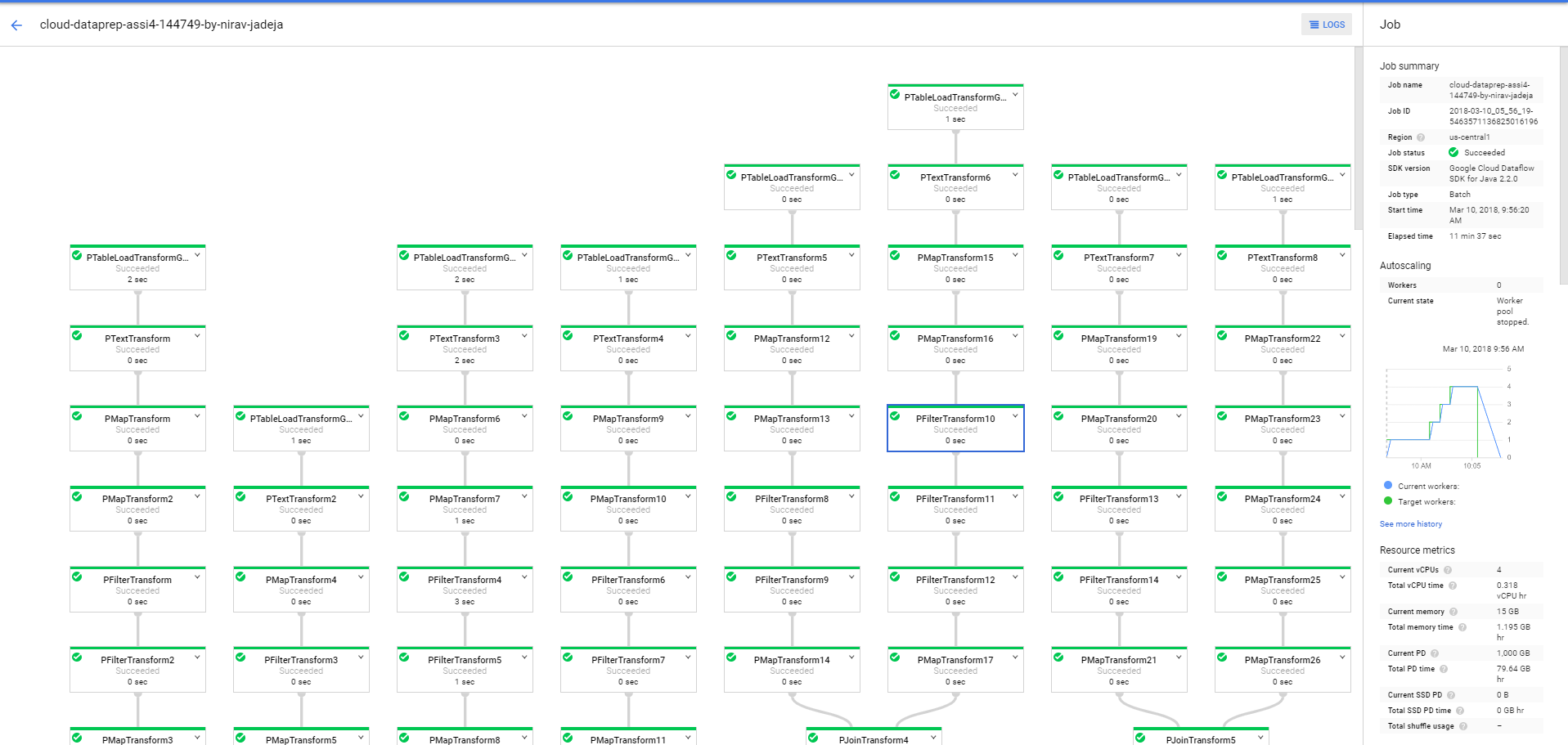
****

Figure 10: Data Flow Diagram, Google Cloud – Part 1

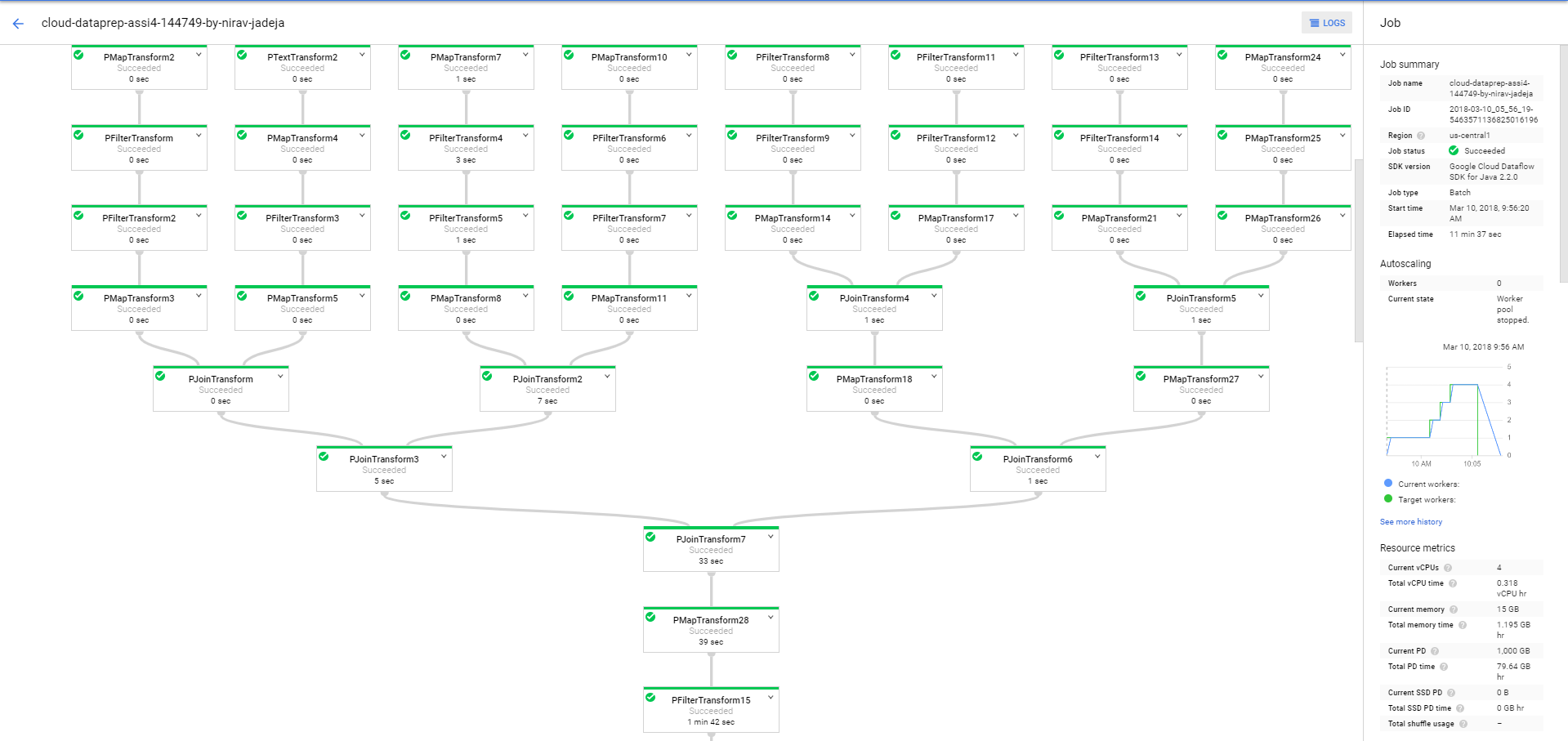
****

Figure 11: Data Flow Diagram, Google Cloud – Part 2

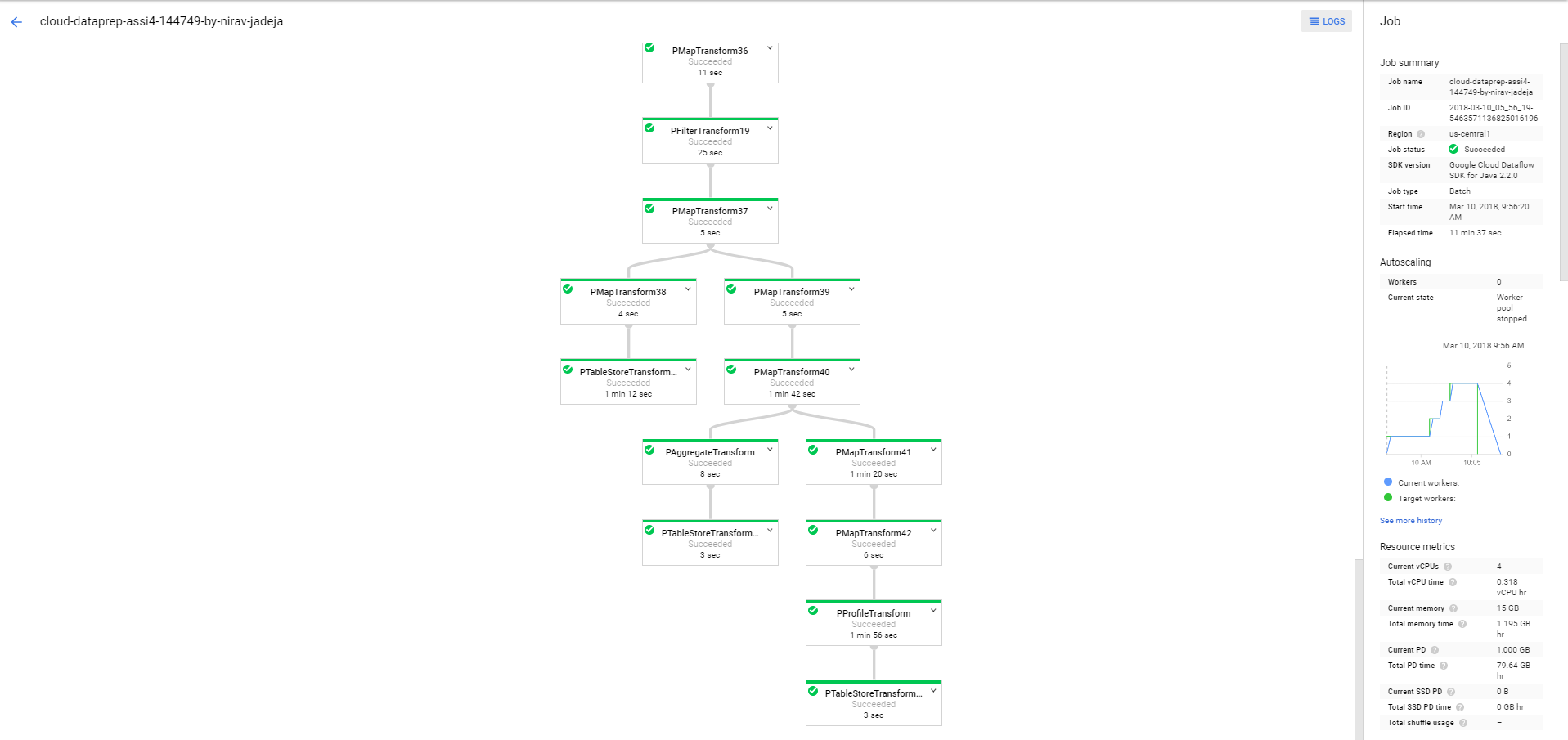
****

Figure 12: Data Flow Diagram, Google Cloud – Part 3

**3.3 DW Schema Diagram**

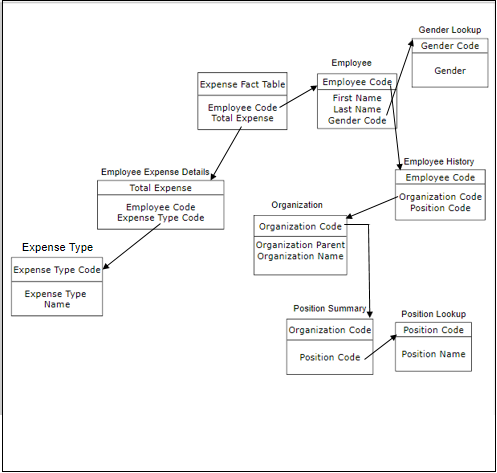


Figure 13: Data Flow Diagram, Google Cloud – Part 3

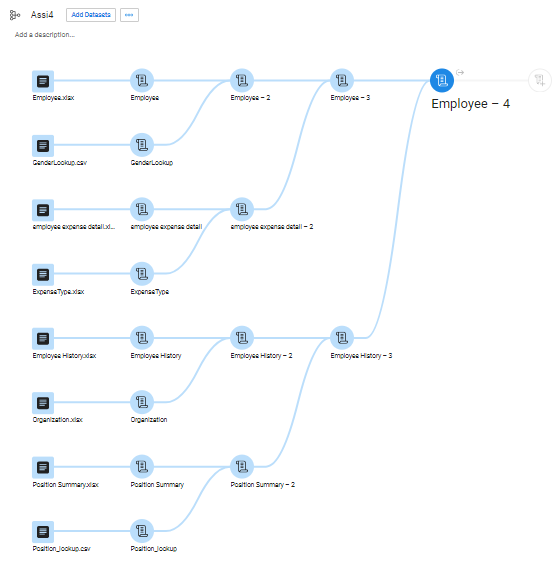


Figure 14: Data tables Joining Diagram

**3.4 Final Dataset**

It has been seen from the final dataset obtained that it has 4.689M rows and all of which are valid with no mismatched data at all. It took 12 minutes for the job to run and given the final dataset.

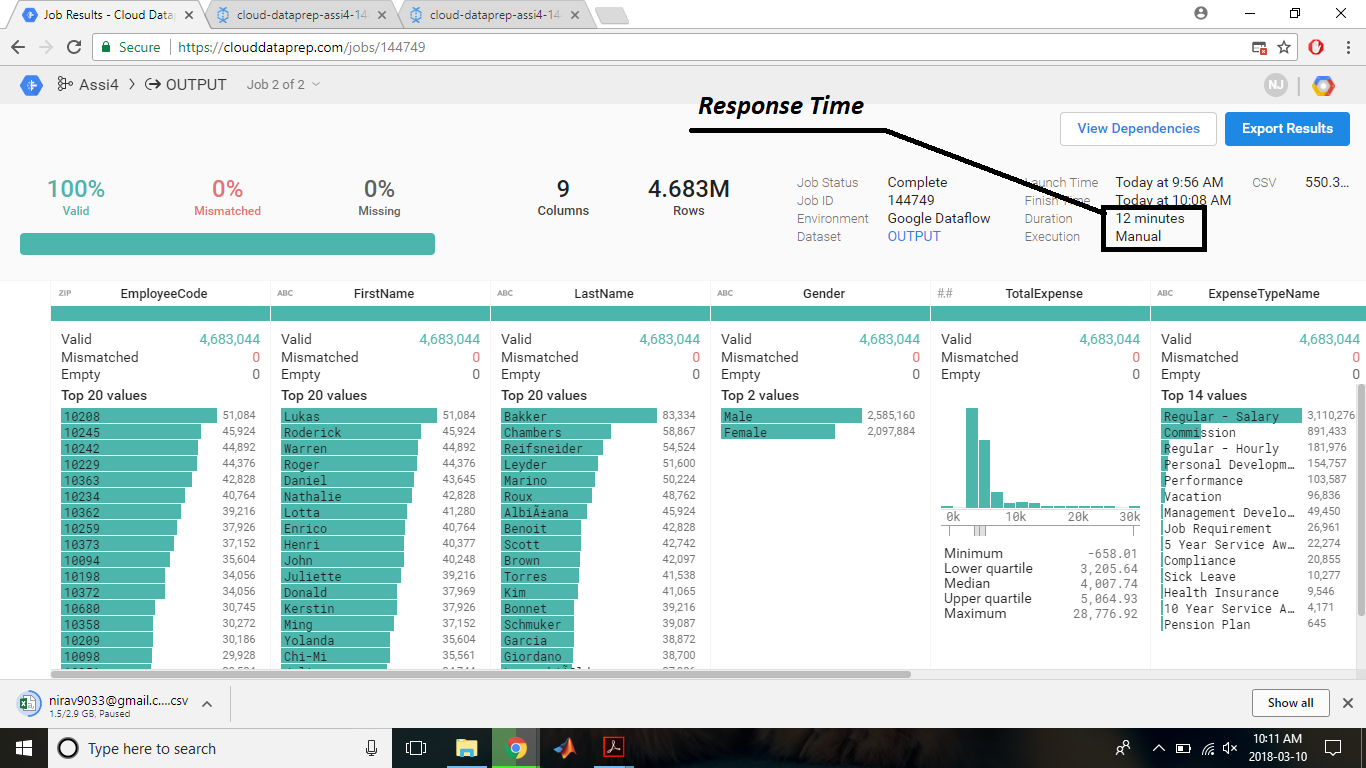
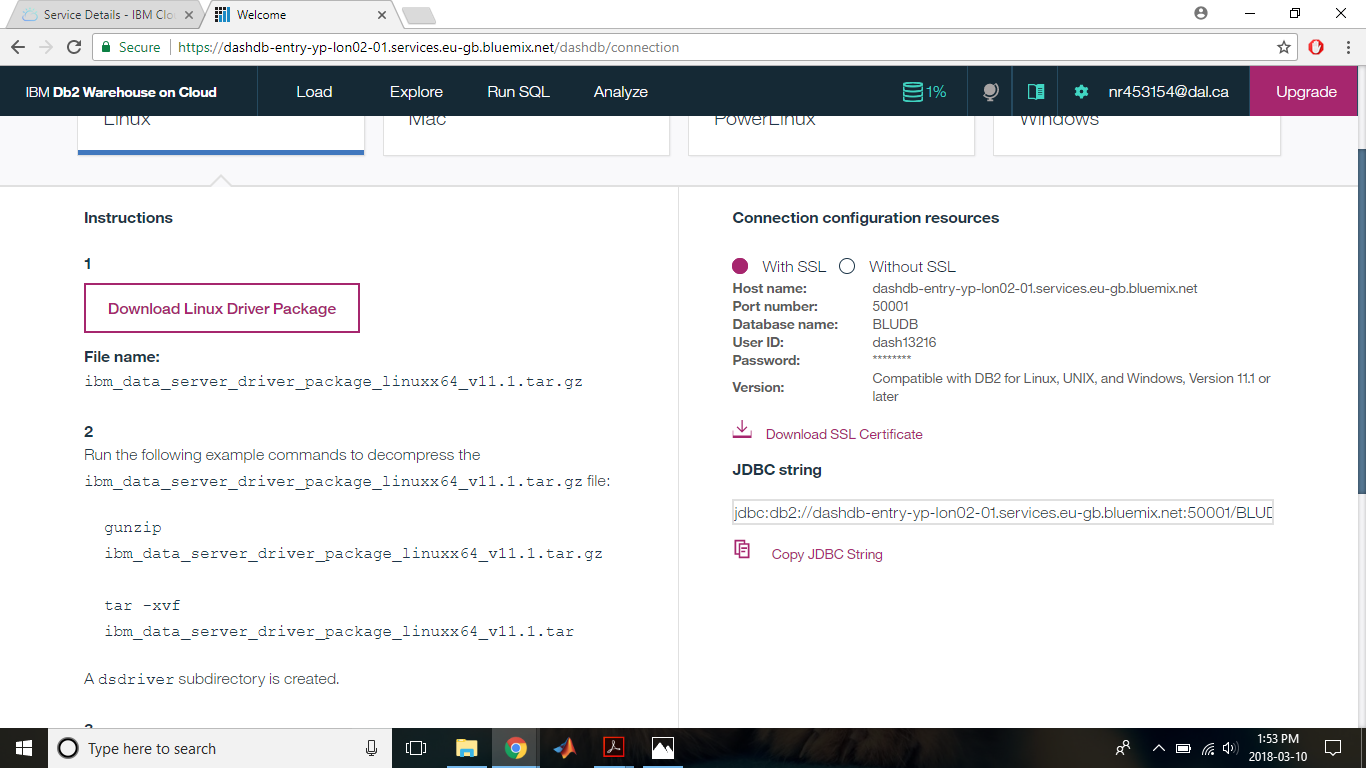
****

Figure 15: Final Dataset

Once we obtain the final dataset and upload it to DB2 warehouse, we establish a connection in the IBM Cognos *[6]* through JDBC string which would help us to load our data from DB2 to implement the queries.

****Figure 15: JDBC Link of uploaded dataset on IBM DB2 Warehouse

1. **OLAP Queries & Reports**

We have used the GOSALESHR schema in the application. It consists of expense details of individual organizations as well as expense details of different positions in the organization throughout the years. With the help of it we made a dataset for our application and loaded it to IBM Cognos for the implementation of OLAP queries.

**Query 1:** What is the total expense by organization?

OLAP: This is an example of roll up OLAP query. Hierarchically organization are at the topmost level hence it is known as **roll up**.

Employee Position ----------> Organization Department ----------> Organization

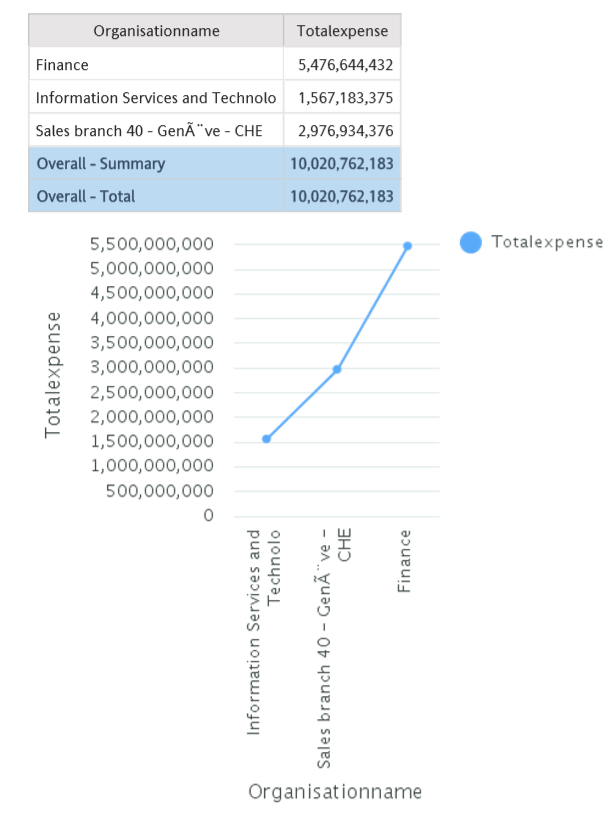


Figure 16: Roll up Operation with Chart

**Query 2:** What is the total expense by position?

OLAP: As position of an employee in any department is at the lowest hierarchy, this is the example of **drill down**.

Employee Position <---------- Organization Department <---------- Organization

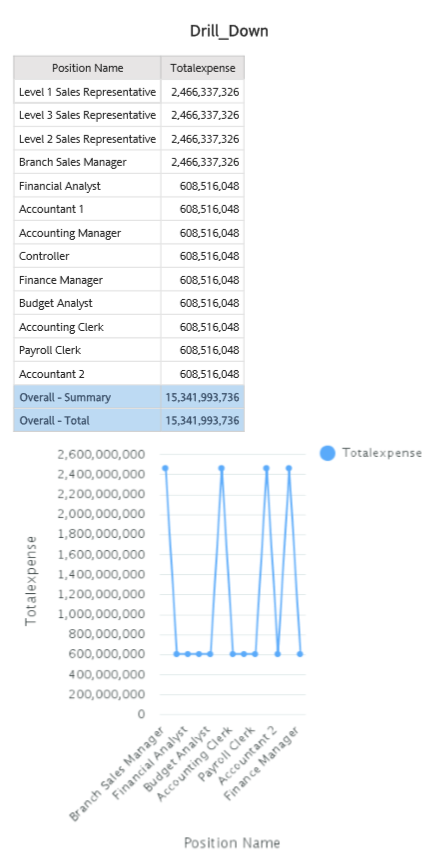


Figure 17: Drill Down Operation with Chart

**Query 3:** What is the total expense in an organization for a position?

OLAP: As the attribute Position is further sliced to find the total expense of an employee position with the help of a custom filter, this is an example of **Slice**.

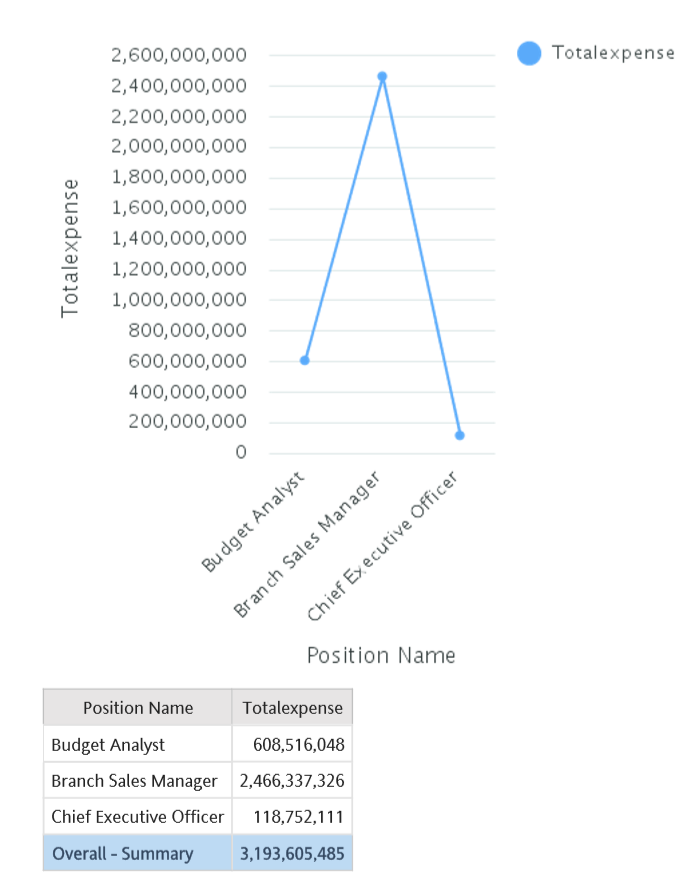


Figure 18: Slice Operation with Chart

**Query 4:** What is the total expense of the compared organizations in a particular type of expense?

OLAP: Here the organization field is diced to compare the desired organizations which is then further sliced in context of expense type hence making it **Dice** OLAP operation. Two custom filters have been used, one for organization and the other for expense type.

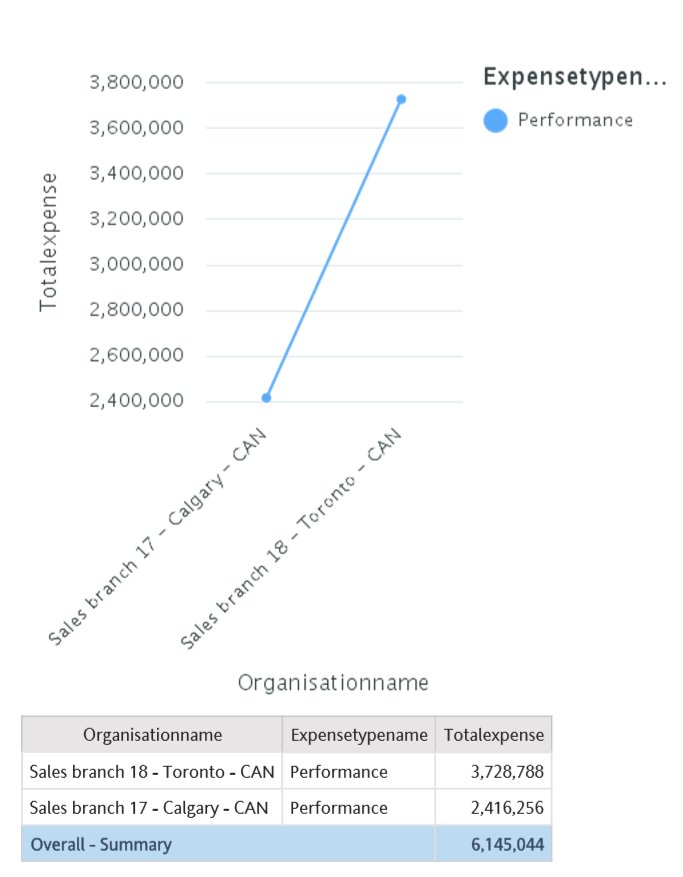


Figure 19: Dice Operation with Chart

The filters and the queries in the report format with the output is as below:

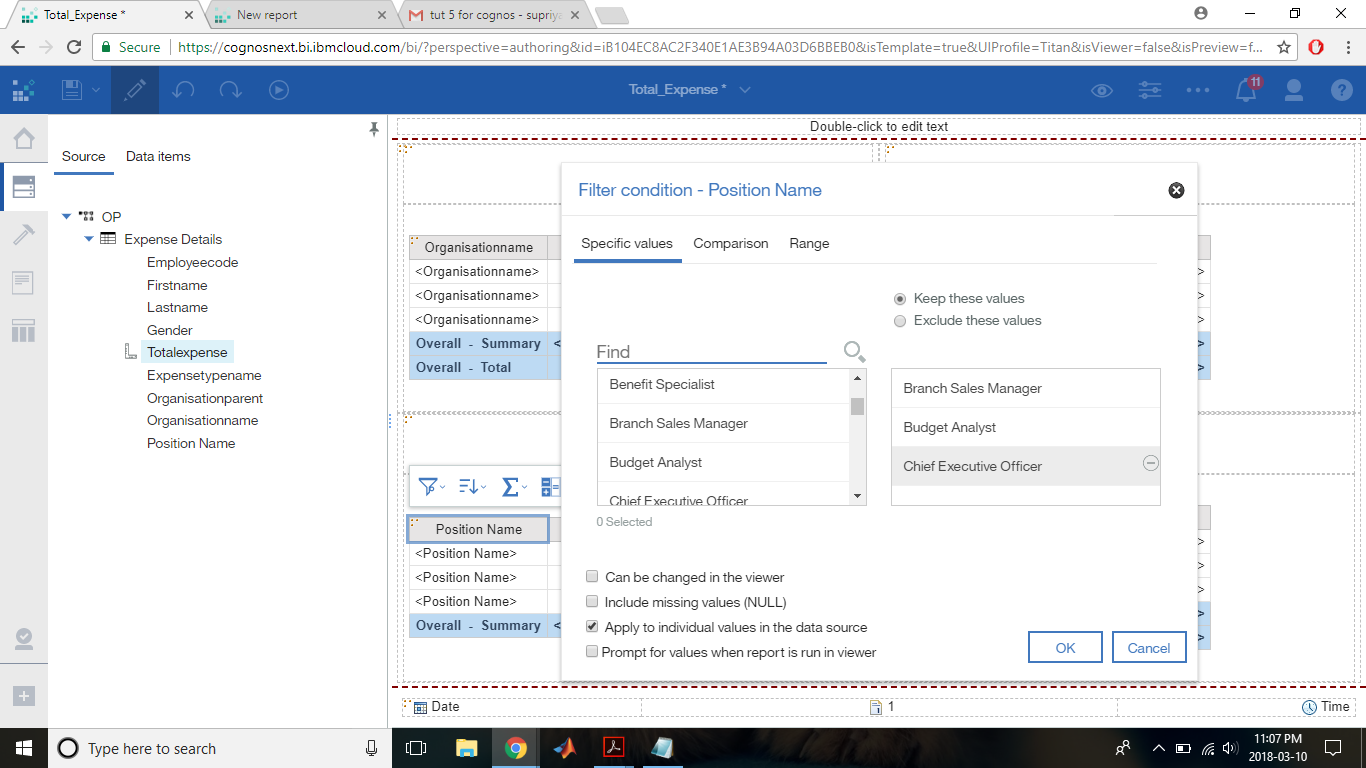


Figure 19: Filter condition used during slice

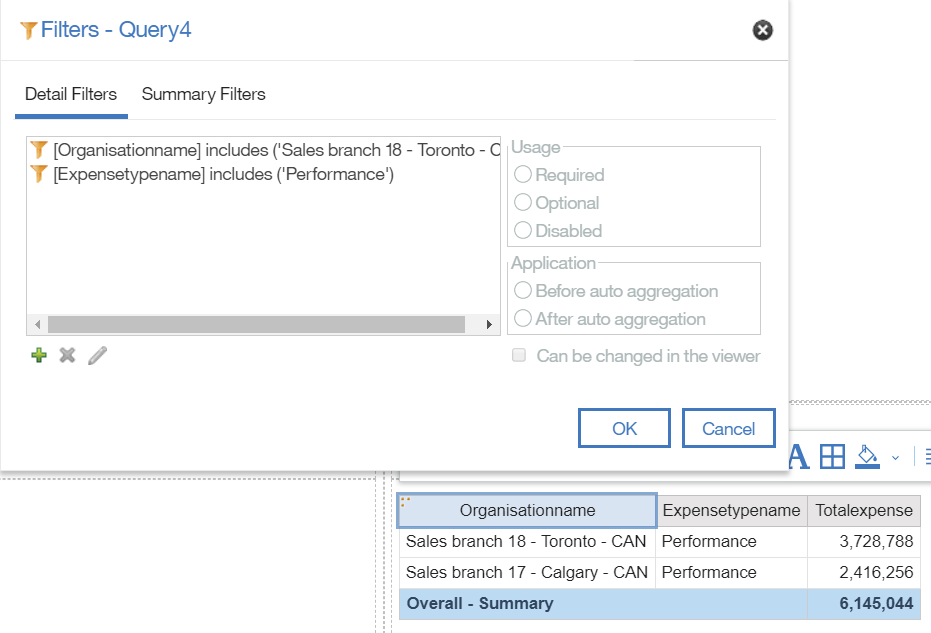


Figure 20: Filter condition used during Dice Operation

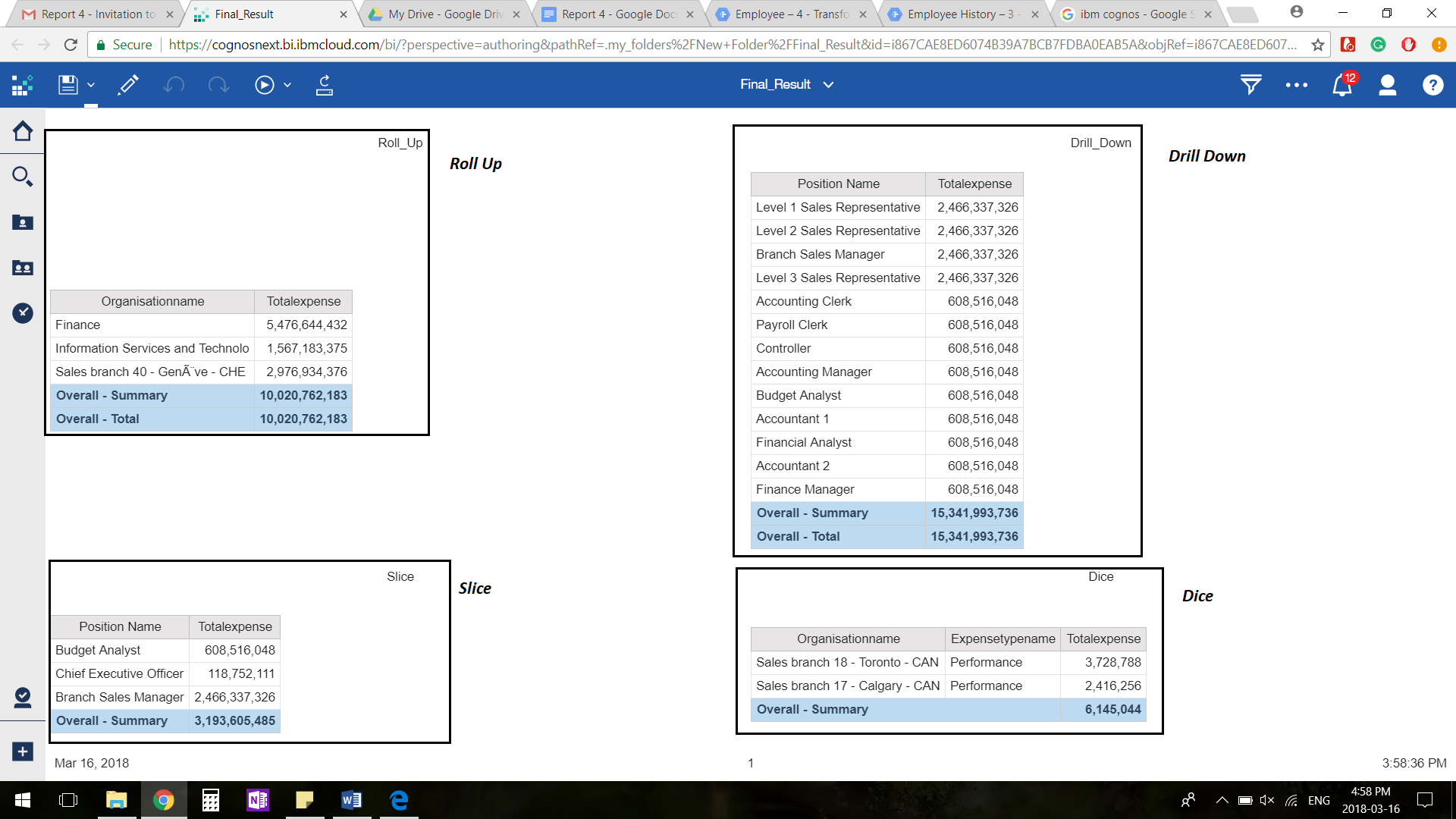


Figure 21: The Result

1. **Summary**

In this assignment, we have created an Organization Expense System Application. First, we have selected dataset from the IBM DB2 warehouse. We removed unnecessary columns from all the tables to perform the joining operation. Further, we have used “Data Prep” Module from Google Cloud to perform ETL operations. In which, we have cleared missing values from all tables and then performed join operation on the tables according to our application. At the end, we have created a dataset which will be used later for performing OLAP operations. Later, we have exported the data in the form of excel file and uploaded it to IBM DB2 Warehouse and connected it to IBM Cognos (using JDBC String) to perform Business operations. Further, we performed various OLAP operations such as Roll-up, Drill-Down, Slice, and Dice by applying filters on respective stages. At the end we have also formed the charts from the created results for the visualization purpose.

**5.1 Experience with Tools:**

It is always a pleasure to try your hands on new tools. We had a great experience working with the Google Cloud Platform and IBM Cognos. Though we were facing some issues with the uploading the dataset (due to its large size) on IBM Cognos. Later, we solved it by applying some changes such as removal of some columns. IBM Cognos is a very helpful tool which handles complex operation and gives results in intuitive way (visualization). Any one can use it and performs various data operations without writing any SQL *[7]* query which is one of the advantage of using it.

**5.2 Links:**

1. Dataset: [Link](https://drive.google.com/open?id=1e10zFETg-EDosqm2blUYlcip4T0To8aA)

2. The Final Result with Graph: [Link](https://drive.google.com/open?id=1rab_2g1dCjJT0brvkR0zXNdzWF4CF_p4)

1. **References**
2. “IBM Inc.” IBM DB2 Warehouse, [Online]. Available: https://www.ibm.com/cloud/db2-on-cloud [Accessed 9th March 2018]
3. “Wikipedia”, “Online Analytical Processing”, [Online]. Available: https://en.wikipedia.org/wiki/Online\_analytical\_processing [Accessed 6th March 2018]
4. “Wikipedia”, “Extract, Transform, Load”, [Online]. Available: https://en.wikipedia.org/wiki/Extract,\_transform,\_load [Accessed 6th March 2018]
5. “Google Inc”, “Google Cloud”, [Online]. Available: https://cloud.google.com/ [Accessed 08 March 2018]
6. “IBM Inc.”, “IBM Cloud”, [Online]. Available: https://www.ibm.com/cloud/ [Accessed 06 March 2018]
7. “IBM Inc.”, “IBM Cognos”, [Online]. Available: https://www.ibm.com/ca-en/marketplace/business-intelligence [Accessed 10 March 2018]
8. “Wikipedia”, “SQL”, [Online]. Available: https://en.wikipedia.org/wiki/SQL [Accessed 07 March 2018]