Programming ETL Scripts

For ORD operations

A Capstone Project Report

Submitted by,

R Santosh Srinivas

Udaya Bhaskar Vaddi

In partial fulfillment of the

Plaksha Technology Leaders Program Requirements

Work done under the guidance of Ankit Acharya, Kedhar Natekar and Rahul Mathur of Piramal Capital & Housing Finance Ltd.

Acknowledgements

We got this project as a part of Capstone Internship programme. We would like to thank Plaksha for providing this opportunity and an open environment where reaching out is easy and communication is encouraged. This work became possible with contributions from a lot of people from the Piramal and Plaksha ecosystem. Below is a non-exclusive list of mentions. Ankit Acharya, our immediate mentor, who helped us throughout the project whose continuous inputs have been critical and formative towards the project. Rahul Mathur and Kedhar Natekar, our senior mentors, who helped in identifying the problem statement and learn in a structured way. Fellows from our batch who helped us with several conversations and inputs on how to measure the learning outcomes.

PLAKSHA TECHNOLOGY LEADERS PROGRAM

Table of Contents

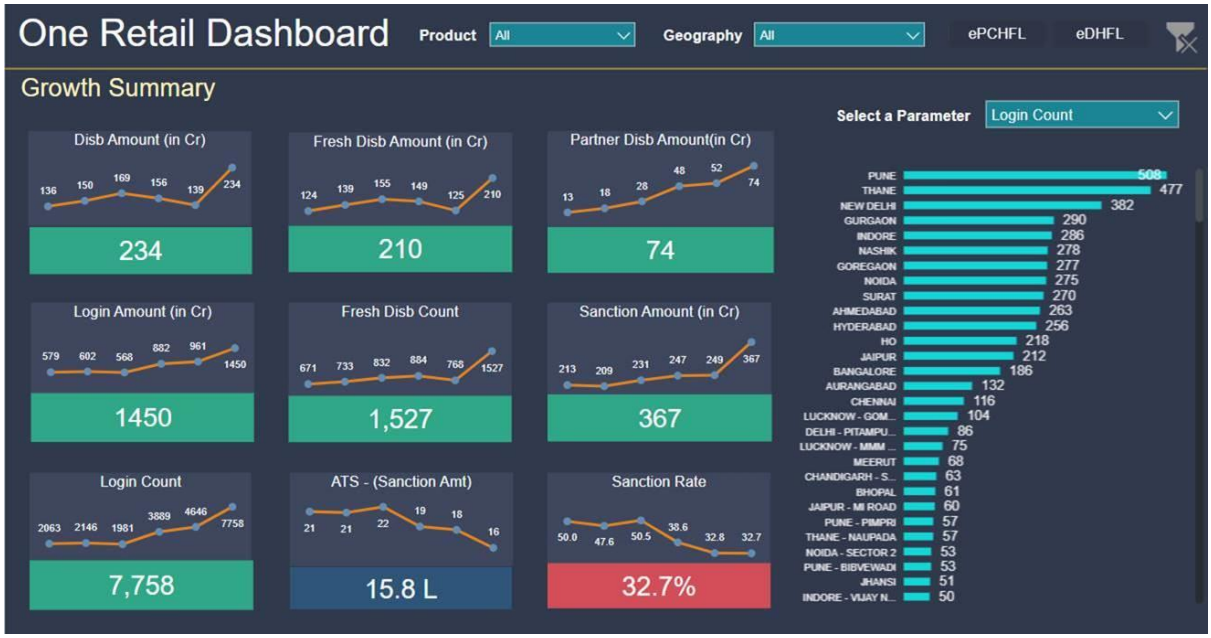
PLAKSHA TECHNOLOGY LEADERS PROGRAM

1. Introduction

One Retail Dashboard (ORD) contains the business insights of Piramal Capital & Housing Finance Ltd. in the form of visualizations. This dashboard is required by the stakeholders (branch managers and cluster/zonal heads) to get a quick insight of how the overall performance of a particular branch/cluster/zone has been, and how they can take necessary actions as applicable.

Below attached is a snapshot of the ORD Dashboard displaying growth summary details such as the loan sanctioned amount, disbursed amount, login count etc for different locations and geographies.

Fig 1: ORD Dashboard



2. Problem Description

Our project in the data engineering team was to create and execute AWS Glue scripts to generate new tables in the Snowflake database required for ORD visualizations. Thus, new tables were added to both UAT (used primarily for testing purposes) and PROD environments (mainly for production purposes) in the Snowflake database.

3. Literature Review

During the initial phase of our internship, our mentors conducted both introductory and knowledge transfer sessions about the company. The introductory sessions introduced us to the business of Piramal by talking about its different loan products: housing, MSME, and used car loans. Recordings of past sessions provided an overview of the different stages of the process by which a loan is issued, checked, and disbursed.

Before the main project was assigned to us, we were trained in the different tools and technologies that would be required such as Python, Snowflake, Pyspark, AWS Glue, and AWS S3.

Snowflake:

We were introduced to Snowflake through training sessions that covered a variety of queries, including select, filter, and table joins. Daily reports in the company were also asked to be given to the Business Intelligent Unit such as the loan pay schedule from uno_ds schema etc. Snowflake queries were written to produce results for the reports and then exported as excel spreadsheets.

Pyspark:

Introduction session on Pyspark and executed basic tasks given such as select commands, filter conditions etc.

ORD sessions:

Knowledge transfer sessions on the dashboard maintained by the sales team including each feature such as the details of the loans in different stages - login, disbursed and sanctioned, targets of the company and comparison of various metrics with the past, current and future.

AWS Glue sessions:

Introduction to AWS Glue where scripts would be written to create tables in UAT and Production environment.

In this way, all these introductory sessions were completed before the project began.

4. Procedure

The following is the steps for the procedure of the project:

a. Segregate the tables into different levels based on dependencies.
b. Set up the environments and required parameters for AWS Glue scripts.
c. Creating AWS Glue Scripts according to the existing Snowflake SQL queries for the existing task.
d. Executing the scripts until they are successful and making necessary changes to make it work.

5. Design and Implementation

Segregation of tables:

All the new tables that had to be migrated in the UAT and PROD environment were given. Each new table was a joint of previous existing tables and other newly created tables.

First, the tables were segregated into 5 different levels based on their dependencies. The higher-level tables depend on the lower level tables such as level 5 tables depend on level 4 and other lower level tables, level 4 on level 3 and lower level tables and so on. Thus, AWS Glue scripts for level 1 tables were created first, followed by level 2 and so on.

The following is the distribution of number of tables at each level:

Level 1: 12

Level 2: 4

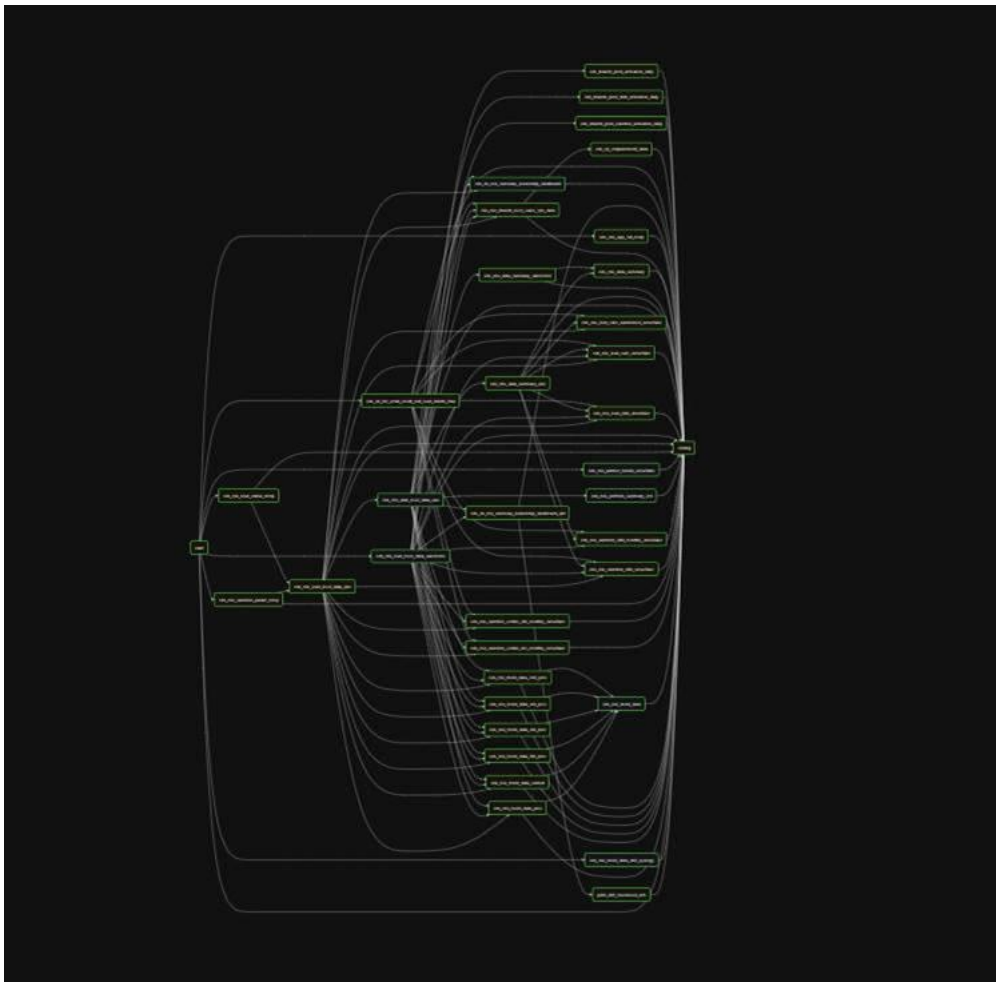
Level 3: 7

Level 4: 13

Level 5: 9

Note: **The names of the tables in the below figure are blurred out due to the lower resolution of the snap, it is strongly advised to not enhance this image at any cost using any sophisticated software as it will have legal repercussions from Piramal Capital and Housing Finance Ltd.**

Fig 2: Dependency Diagram



Setting up AWS Glue environment:

1. Snowflake DB Parameters: Each AWS Glue script needed to be connected to the correct schema in the snowflake database to migrate the data. For this connection to be established parameters in the form of dictionary key value pairs are sent including the schema and location link etc.

2. Script run time: Each AWS Script is allotted a maximum run time to save costs and to get out of situations where the code is running for too long in dire cases such as poor coding (multiple cartesian joins for ex). The maximum run time for the UAT environment scripts was set to 5 minutes and for PROD was set to 10 minutes.
3. Maximum number of machines: Each script is allocated a maximum number of DPUs for simultaneous execution to manage resources efficiently. The maximum number of machines for UAT and PROD was set to 5 and 10 respectively.

AWS Glue script execution:

Once the AWS Glue Script configuration is set, we begin writing the script. Pyspark clusters are used as intermediaries between the AWS Glue script and the snowflake database. The base tables required for the creation of our new table are loaded into the Pyspark cluster. Then Pyspark SQL query is written performing joints and manipulation to load the data into our required table in the Pyspark cluster. Then the data in the required table is migrated from the Pyspark cluster to the Snowflake database. Then the Pyspark query is written by conversion of the given SQL query which involved the following steps such as removing " to `, casting of dates, casting of data types etc. The query was refined continuously while debugging for the changes. This was the essence of writing the AWS Glue script and the most challenging part of the project.

Data Validation:

Data Validation is done for the migrated data in the UAT and PROD environment. UAT data validation includes checking the data type of all the columns, checking if the rows are not repeated with the help of unique identification of rows using primary key columns etc.

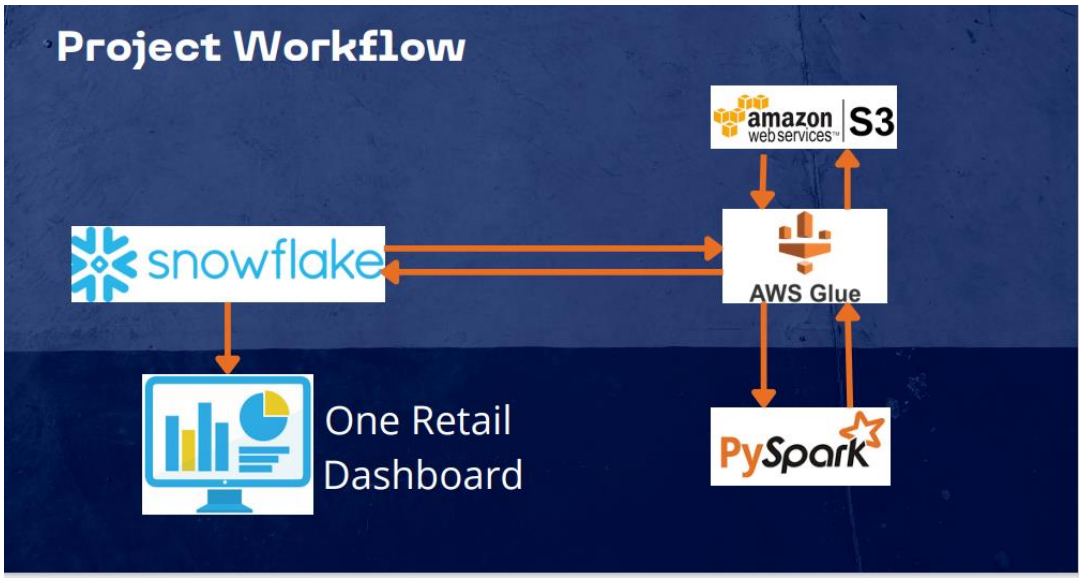
Once the data is validated for the UAT env, PROD env data is validated. First, it is checked if the number of rows in UAT data and PROD data is the same. If so, the two data frames are compared using the pandas function for validation.

If the number of rows in UAT is more than that of PROD, there is a high probability that the data in PROD env is incorrect.

If the number of rows in PROD is more than UAT, there is a high chance that new data must have been added since the script was last executed. So, data validation is done for rows which are the same in UAT by comparing directly and for the new rows added, data validation is done accordingly for different cases.

Below is a snapshot summarizing the entire project workflow

Fig: Project Workflow Summary



6. Results Obtained

Total of 45 scripts in UAT and 45 scripts in PROD were executed successfully. The migrated data was then validated which is then used in the ORD Dashboard providing insights to the stakeholders.

7. Conclusion

In conclusion, our internship at Piramal gave us a comprehensive understanding of the role of data engineering in the company. Introduction to the business of Piramal was given with the help of orientation sessions and recordings detailing the products and launches of Piramal. We then learnt about the tools and technologies required for the data engineering project such as writing queries in Snowflake SQL, Pyspark commands and understanding of AWS Glue scripts and environment.

Once the understanding of the tools and technologies was done, we began getting into the main project that involved migration of ORD data into the snowflake UAT and PROD environment for the ORD Dashboard. This involved first segregation of the tables into levels, followed by setting up of the Glue environment, execution of AWS Glue scripts and data validation which was the final step of the project. Besides technical skills, soft skills such as interactions with mentors and completing tasks on time were also acquired. All the expectations and requirements were delivered rightly on time with no tasks pending for the future.