Ecole Supérieure Privée
d'Ingénierie et de Technologies

ODDO BHF

Report

PE-BI

CUSTOMER PREDICTION, PROFILING AND CLASSIFICATION OF ODDO-BHF SECURITIES CLIENTS

Mohamed Iheb Bousnina

Fedi Bayoudh

Khalil Ben Romdhane

Montassar Thabti

Mhadheb Ben Mahmoud

College Year 2019 - 2020

Table des matières

Figure Table

Liste des tableaux

abreviations list

- **BI** = **B**usiness **I**ntelligence
- **CRM** = **C**ustomer **R**elationship **M**anagement
- **DW** = **D**ata **W**arhouse
- **ETL** = **E**xtract **T**ransform **L**oad
- **KNN** = **K**- **N**earest **N**eighbor
- **SA** = **S**taging **A**rea
- **SVM** = **S**upport **V**ector **M**achine

General Introduction

In this data-driven world, Data Analytics has become vital in the decision-making processes in the Banking and Financial Services Industry. Investment banking and other businesses wherein, real-time information is used, volume, as well as the velocity of data, has become critical factors.

Today, data analytics practices have made the monitoring and evaluation of vast amounts of client data including personal and security information by Banks and other financial organizations much simpler.

There are several use cases in which Big Data Analytics has contributed significantly to ensure the effective use of data. This data opens up new and exciting opportunities for customer service that can help defend battlegrounds like payments and open up new service and revenue opportunities.

Business Intelligence (BI) is necessary to compete in today's data-driven marketplace. BI can provide you with meaningful reporting and actionable data that can maximize your revenue, improve efficiency, and deliver better client outcomes, as well as positively impacting your bottom line. Identifying and prioritizing key opportunities is necessary to maximize your financial goals

The goal for a successful BI service is locating, collecting, securely storing and aggregating General Introduction the necessary data elements in a central location, typically the 'cloud' today, and then performing analytics to provide reporting promptly to meet changing marketplace dynamics and client care needs.

Chapitre 1

Project Conception

Plan

Introduction

This first chapter is dedicated to the presentation of the preliminary study which amounts to the first stage of our project titled Securities and Customer's Profiling and Prediction. First, we establish the business objectives that we aim to fulfill by capturing the project's goals. Next, we will introduce the project's context, data source identification and description and the system architecture.

1.1 Enterprise Presentation

Oddo bhf is an independent Franco-German financial services group, with a history stretching back over 150 years. It was created from the alliance of a French family-owned business built up by five generations of stockbrokers and a German bank specializing in Mittelstand companies. With 2,300 employees (1,300 in Germany and 1,000 in France), and more than 100 billion euros in assets under management, Oddo bhf operates in three main businesses, based on significant investment in market expertise : private banking, asset management and corporate and investment banking.

Figure 1.1: ODDO BHF logo

1.2 Project context

The goal of DDO BHF is to provide their customers with financial decisions and stay connected to them so ODDO BHF manage their data and incorporate it into its business and professional practices. Helping ODDO to make those decisions is our first mission and that by presenting prediction values for its customers so we analyze and interpret the data to reach better decisions.

1.3 Business objectives

The Bank's objective is to maximize the revenues that's why we should analyze the provided data and the scrapped external data to categorize the investment per Client/Company, and analyze the log data of the client so we can determine his activities and predict the amount that must be spent by a client so he can achieve good results. Finally keep Regulars checks with the clients so we can insure better deals and continuous investments. To reach this goal, the project must be composed of three different segments :

1.3.1 Scoring and classification

☛ **BI**

— Determining commercial customer segments

— Determining customer profiles by similarity

☛ **Data Mining**

— Determination of an individualized risk aversion score

— Determination of client selection criteria

1.3.2 Prediction

☛ **BI**

— Individualized collection amount

— Individualized amount of outflow

☛ **Data Mining**

— Prediction of business opportunity due + X months

— Potential of cross-selling with other products (life insurance, PEA, ...)

1.4 Data source identification and description

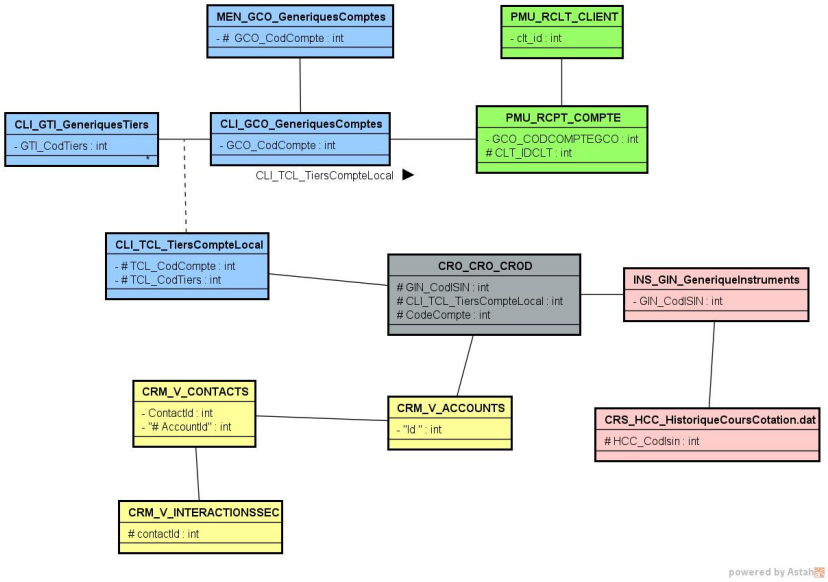

Figure 1.2: Extract of Class Diagram

1.5 System architecture diagram

After detailing and understanding the data we begin with Extracting data from various sources, second transforming the data and finally loading data into a data warehouse. ETL (Extract, Transform and Load) is a process in data warehousing responsible for pulling data out of the source systems and placing it into a data warehouse.

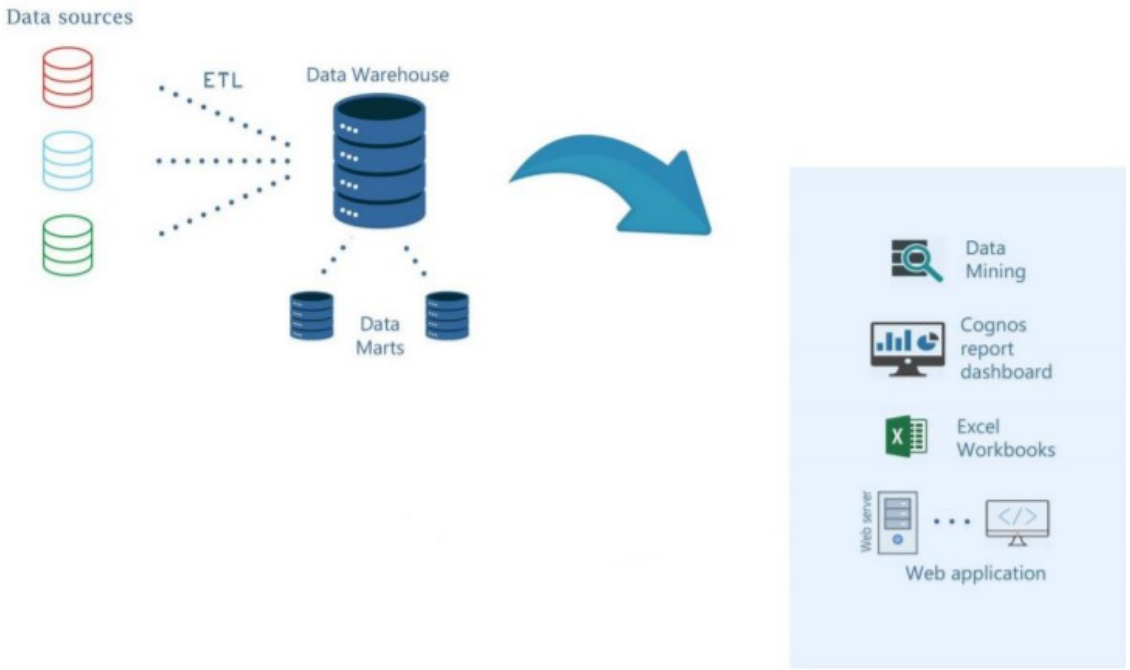

Figure 1.3: System architecture diagram

1.6 DataWarehouse schema

A data warehouse (DW) is designed to support business decisions by allowing data consolidation, analysis and reporting at different aggregate levels. Data is populated into the DW through the processes of extraction, transformation and loading. These technologies help executives to use the warehouse quickly and effectively.

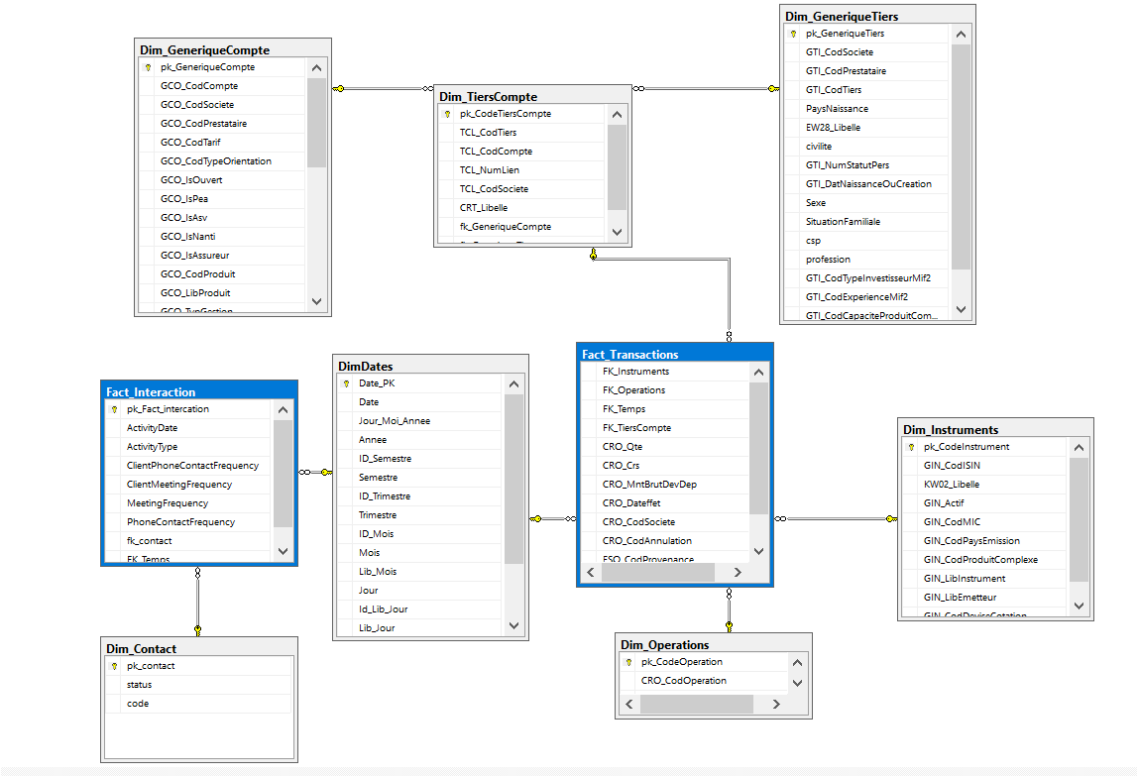

Figure 1.4: Data warehouse diagram

1.7 Conclusion

This first chapter gave us an overview of the general context of our project, we clarified our objectives and the needs of our client Oddo bhf that was done by specifying the objectives of the business and the data source identification.

Chapitre 2

Data Integration

Plan

Introduction

In this chapter, we will move to the process of project implementation by shedding light on the various tools and technologies provided to ensure the accomplishment of this project.

2.1 Data Integration

2.1.1 ETL

Extract-transform-load, known as ETL, is a middleware technology that can perform massive synchronizations of information from one data source to another. Depending on the context, one is led to exploit different functions, often combined between them : "extraction", "transformation", "conversion" ...

2.1.2 Tools and technologies

So we will begin by the Phase 1 which is the data collection and data preparation, In this phase we will use :

Jupyter

Figure 2.1: JupyterLab logo

Jupyter is a web application used to program in more than 40 programming languages, including Python, Julia, Ruby, R, or Scala2. Jupyter is an evolution of the IPython project. Jupyter allows you to make notebooks or notebooks, that is to say programs containing both text in markdown and code in Julia, Python, R ... These notebooks are used in data science to explore and analyze Datas.

Python

Figure 2.2: Python logo

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant 18 white space. It provides constructs that enable clear programming on both small and large scales.

SQL Server

Figure 2.3: SQL Server logo

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network.

Visual Studio

Figure 2.4: Visual studio logo

Microsoft Visual Studio is an Integrated Development Environment(IDE) developed by Microsoft to develop GUI(Graphical User Interface), console, Web applications, web apps, mobile apps,etc. With the help of this IDE, you can create managed code as well as native code. It uses the various platforms of Microsoft software development software like Windows API.

2.1.3 Load of dimensions and facts

A fact table works with dimension tables. A fact table holds the data to be analyzed, and a dimension table stores data about the ways in which the data in the fact table can be analyzed. Thus, the fact table consists of two types of columns. The foreign keys column allows joins with dimension tables, and the measures columns contain the data that is being analyzed.

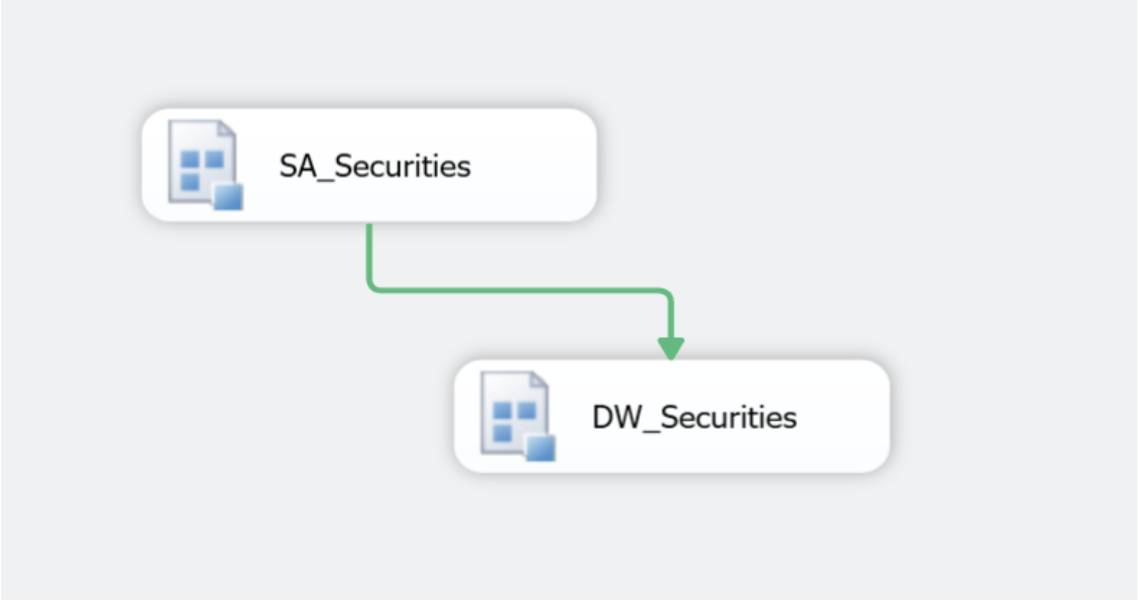

Figure 2.5: Relation between the staging Area and the Data Warehouse

In this part, we used the staging area SA as an intermediate storage used for data processing during the extract, transform and load (ETL) process. The data staging area sits between the data sources and the data targets, which is the data warehouses in this project.

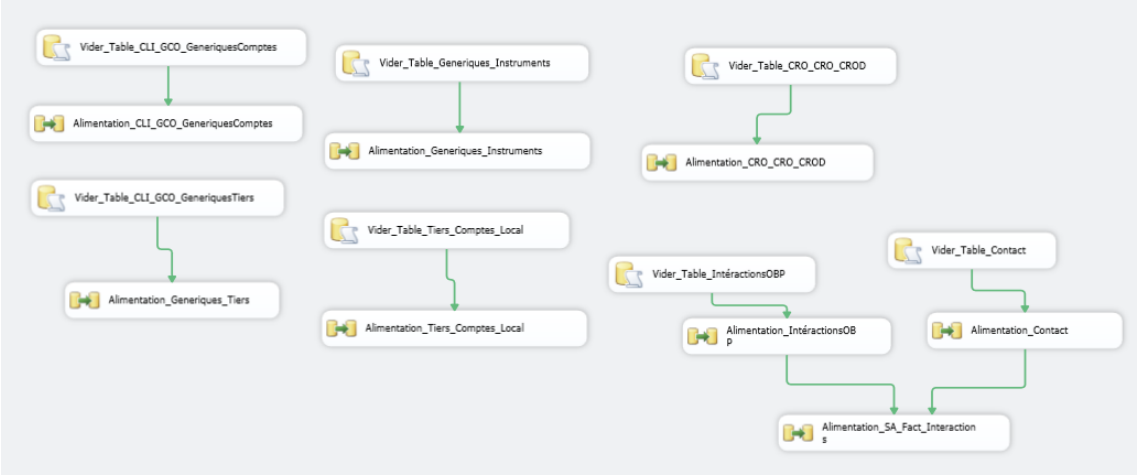

Figure 2.6: Load of Staging Area SA

After the Staging Area, we loaded the dimensions then the Facts.

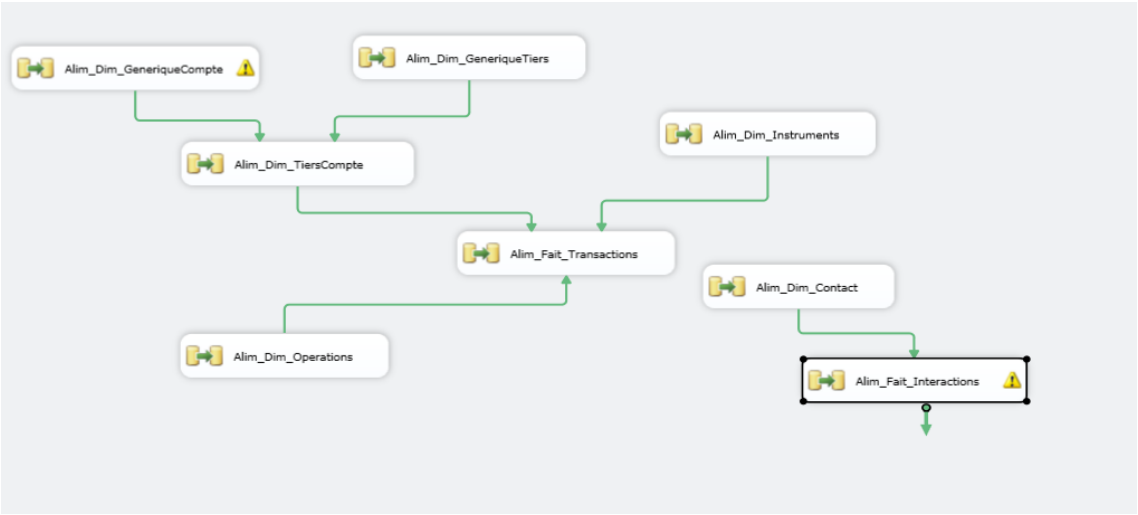

Figure 2.7: Load of Data-Warehouse DW

Dimension GeneriqueCompte

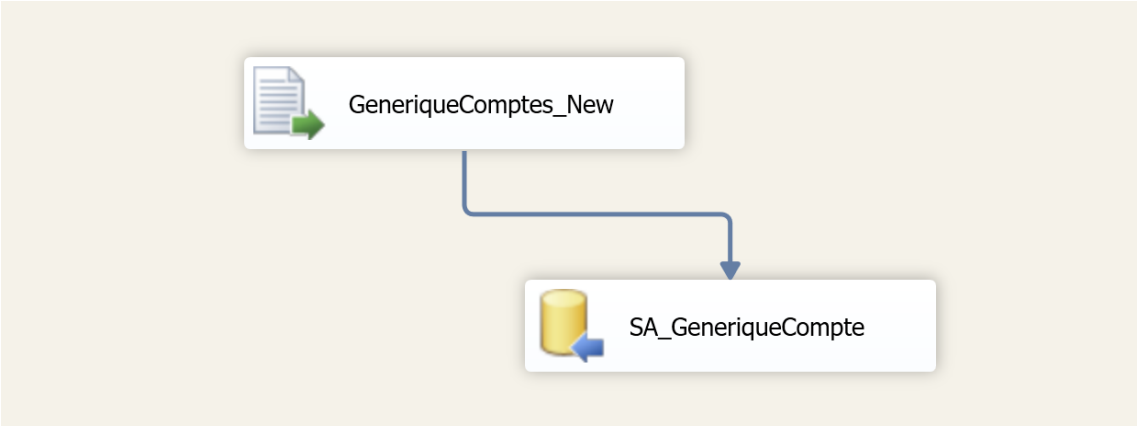

Figure 2.8: Load of GeneriqueCompte SA

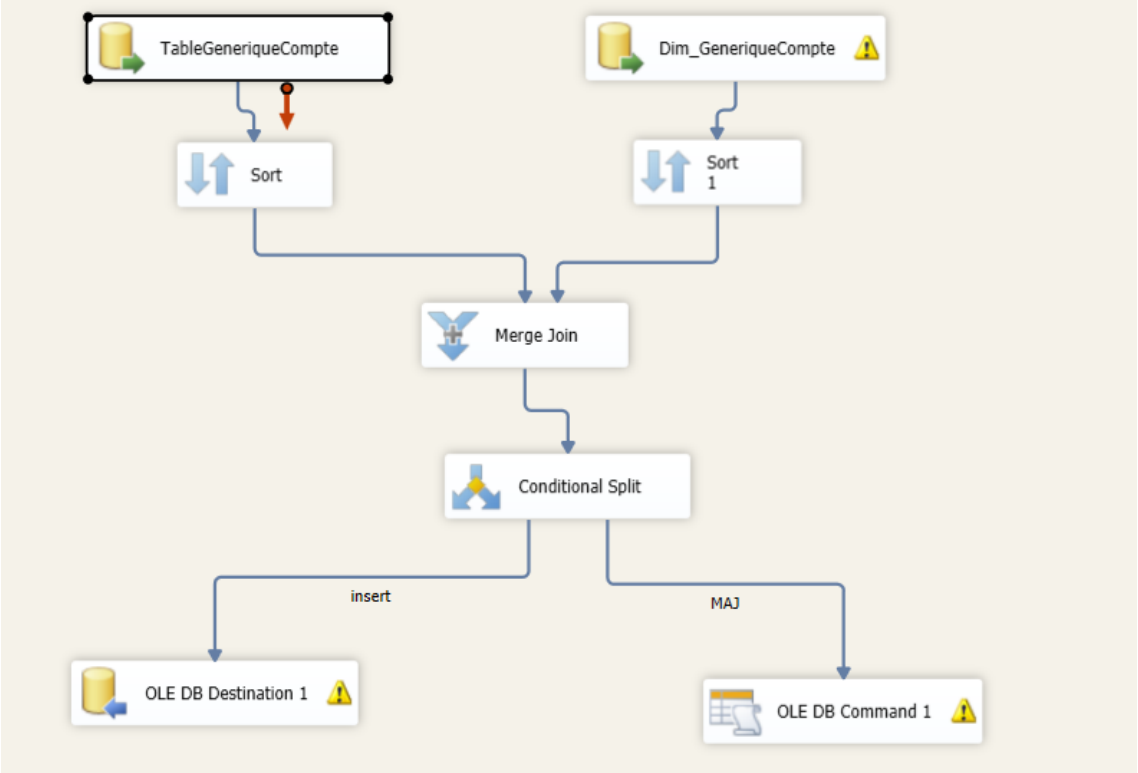

Figure 2.9: Load of GeneriqueCompte DW

Dimension GeneriqueTiers

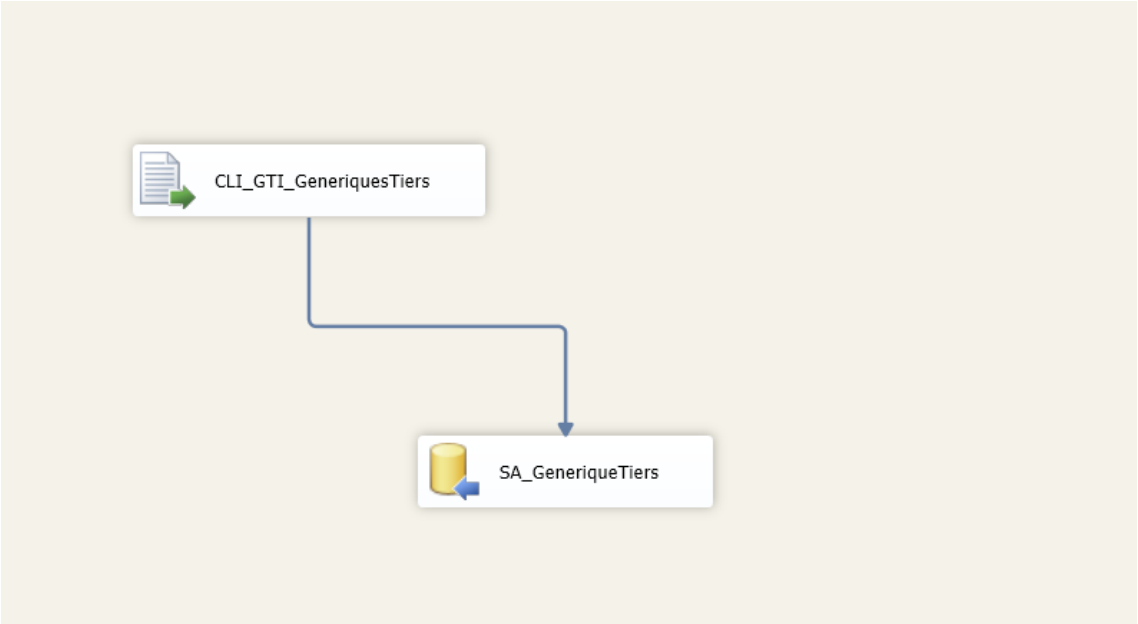

Figure 2.10: Load of GeneriqueTiers SA

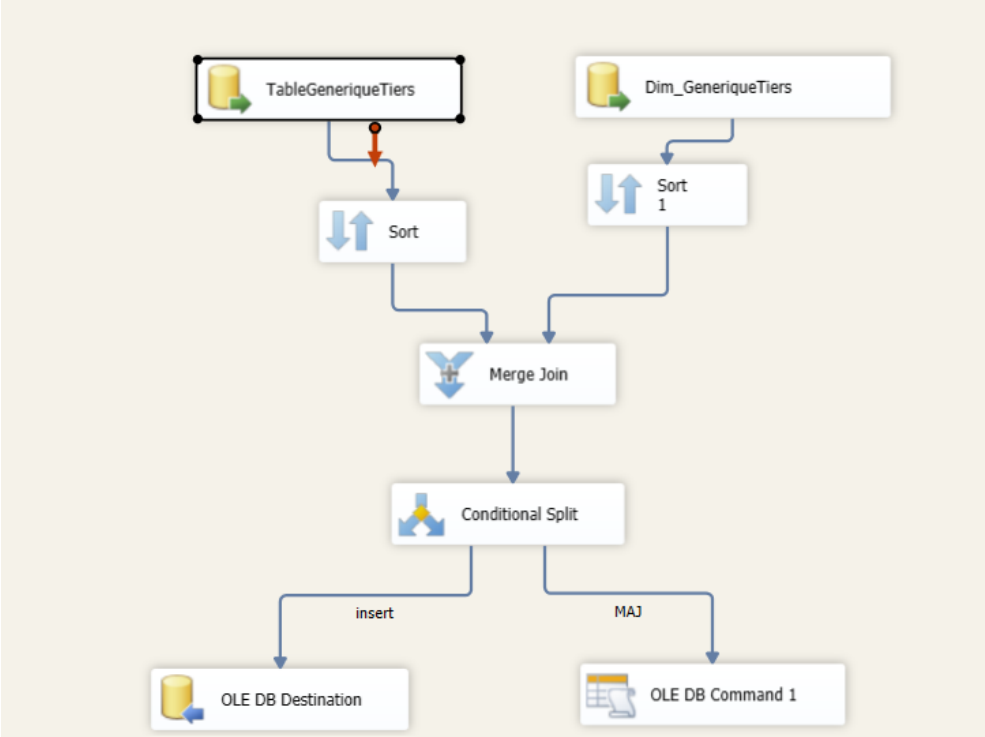

Figure 2.11: Load of GeneriqueTiers DW

Dimension TiersCompte

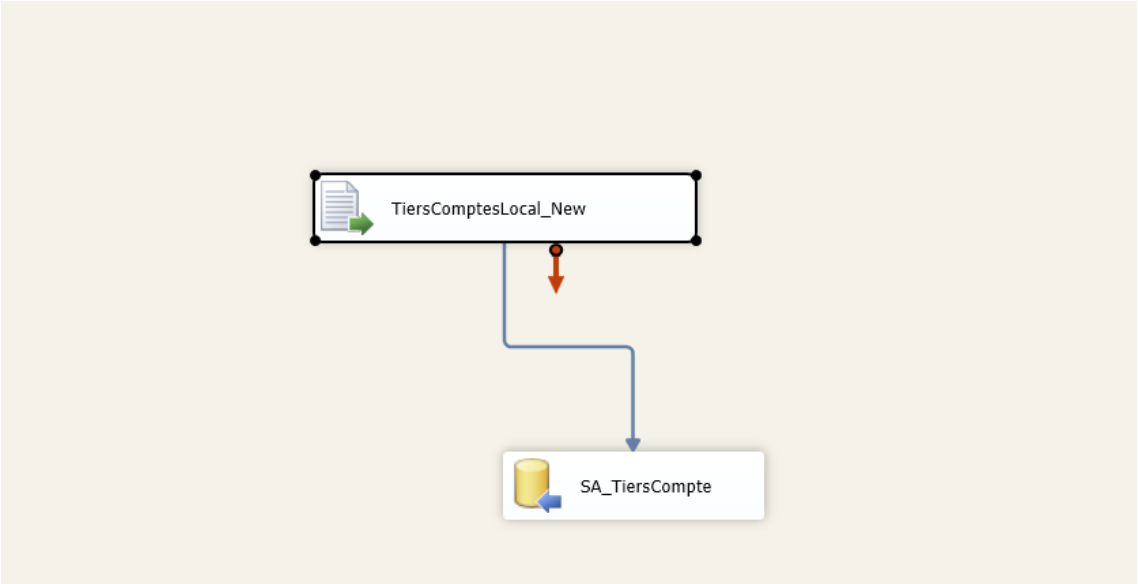

Figure 2.12: Load of TiersCompte SA

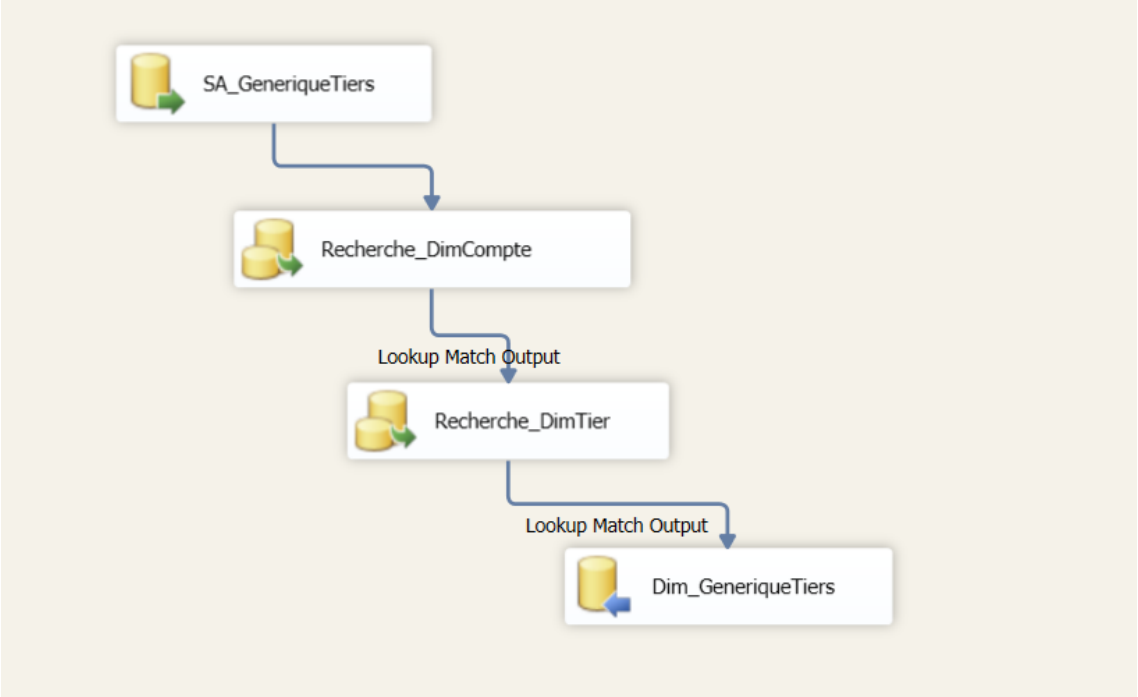

Figure 2.13: Load of TiersCompte DW

Dimension Instruments

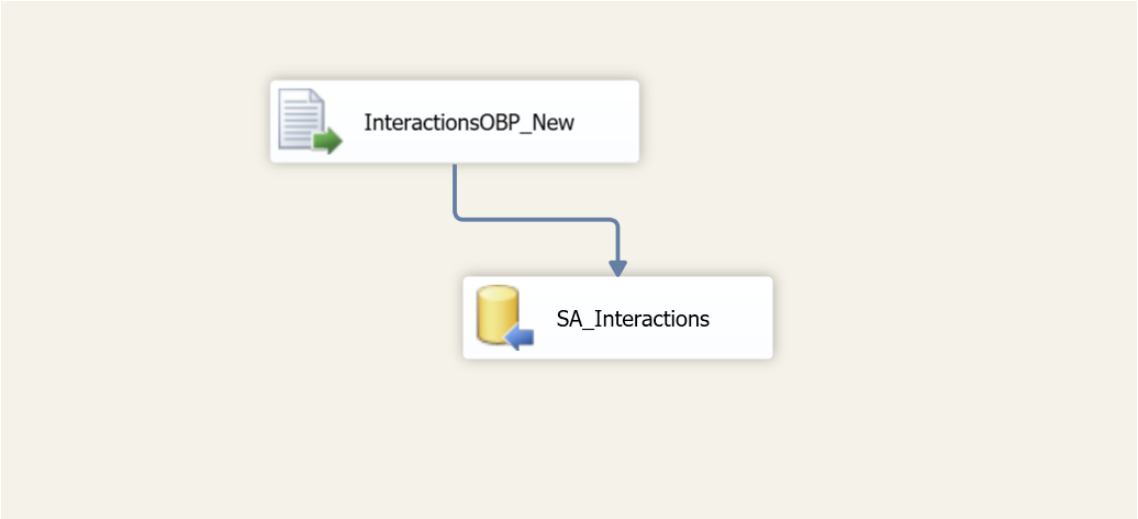

Figure 2.14: Load of Instruments SA

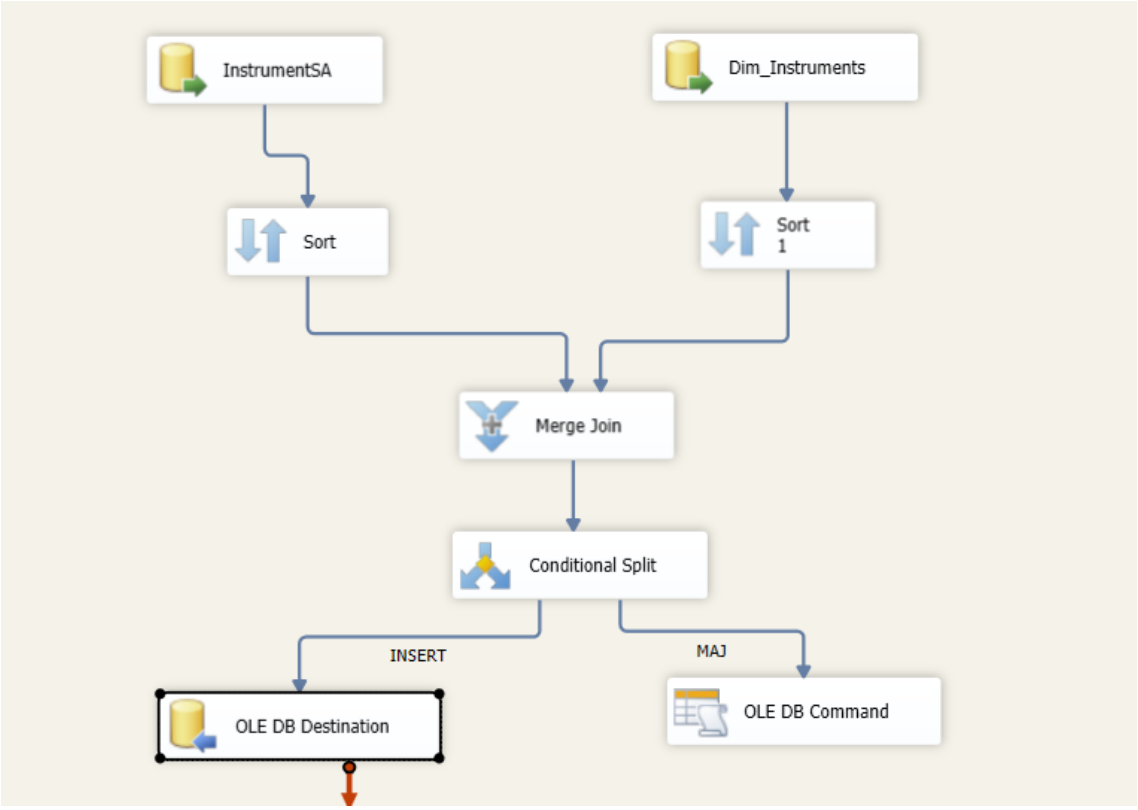

Figure 2.15: Load of Instruments DW

Dimension Operation

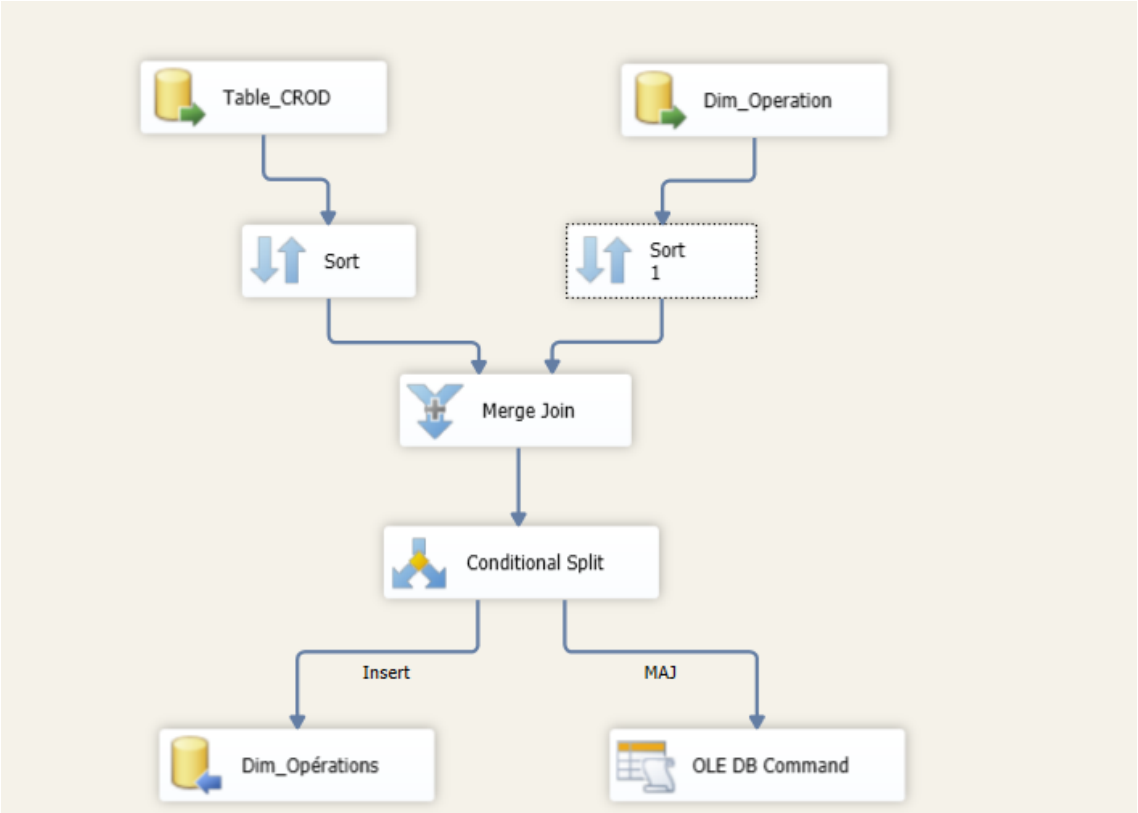

Figure 2.16: Load of Operation DW

Dimension Contact

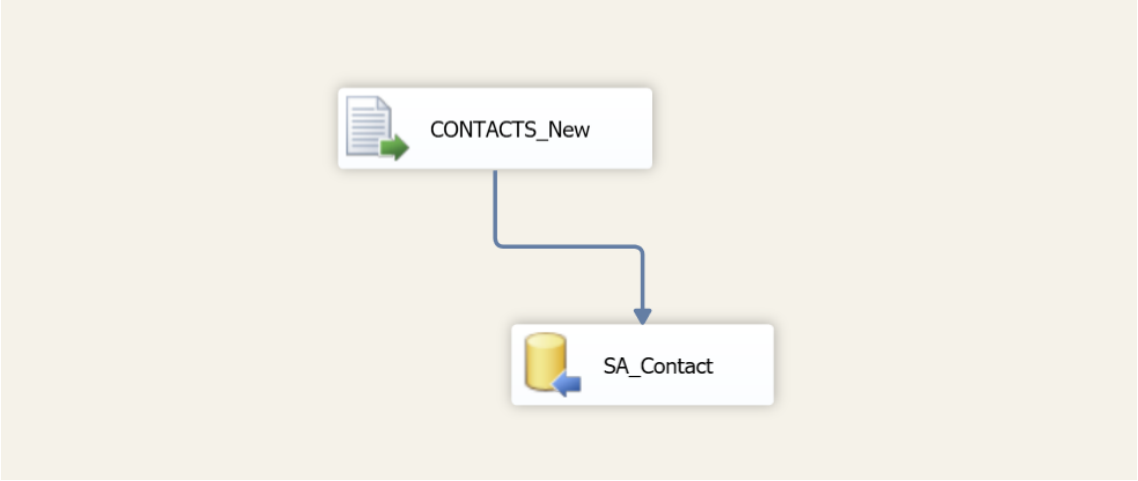

Figure 2.17: Load of Contact SA

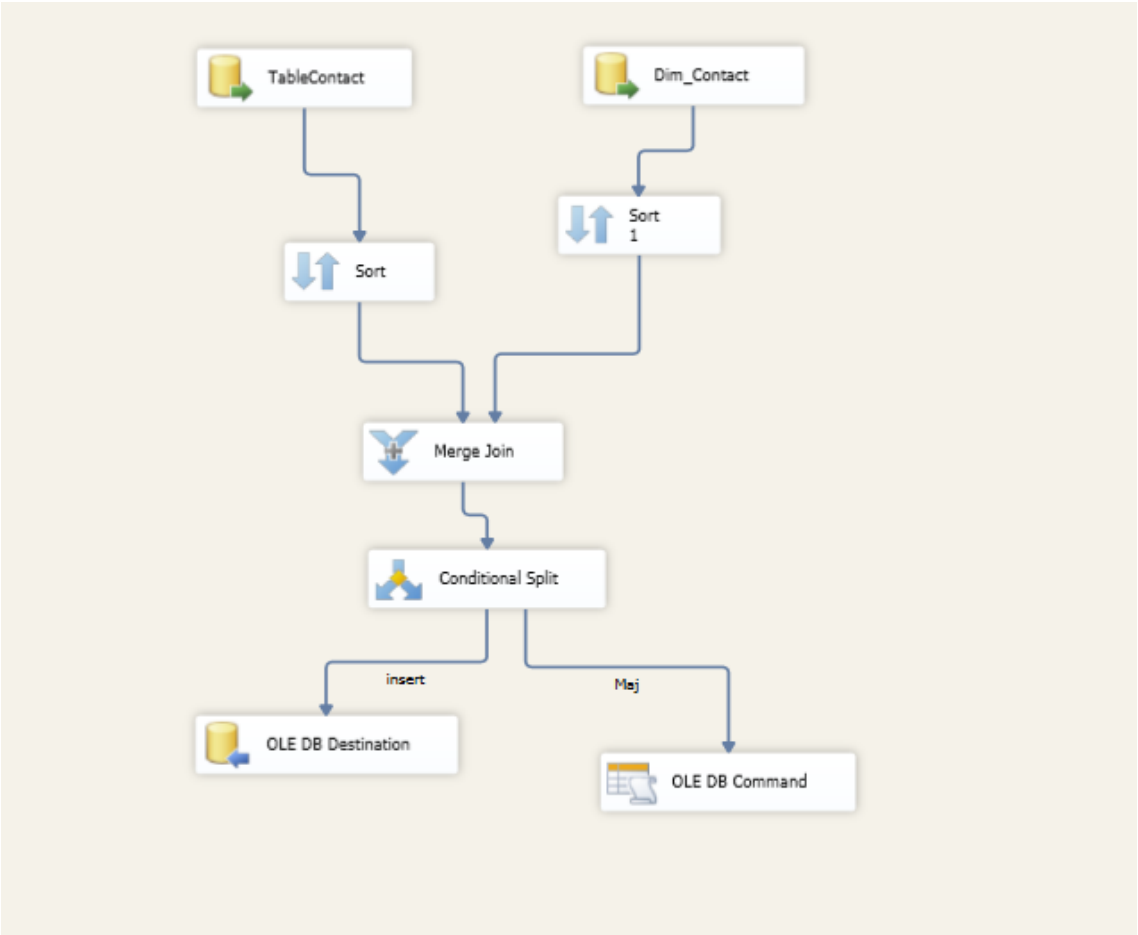

Figure 2.18: Load of Contact DW

Fact Interaction

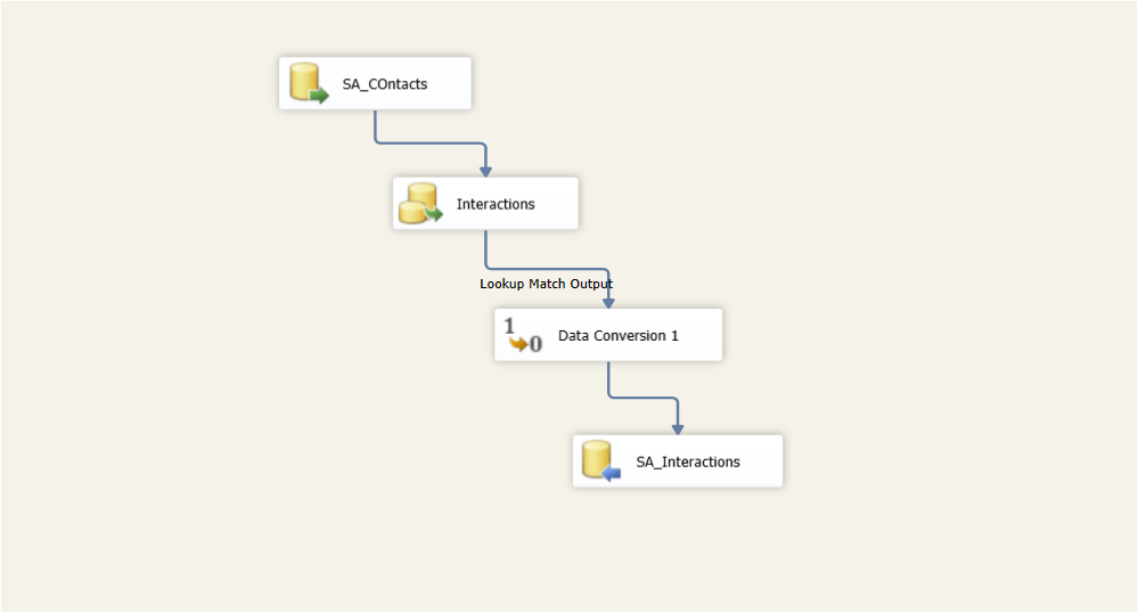

Figure 2.19: Load of Fact-Interaction SA

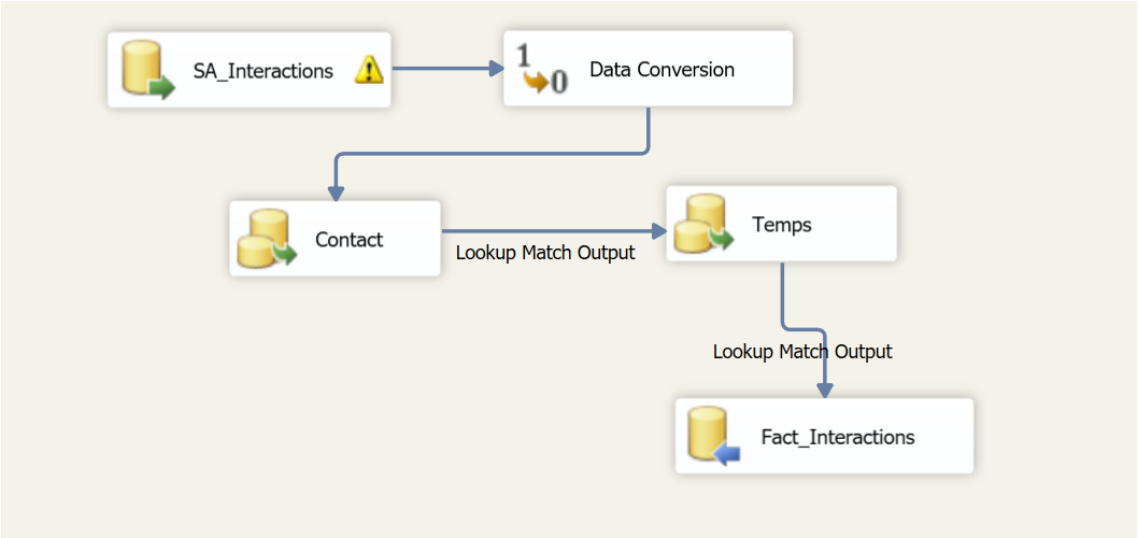

Figure 2.20: Load of Fact-Interaction DW

Fact Transactions

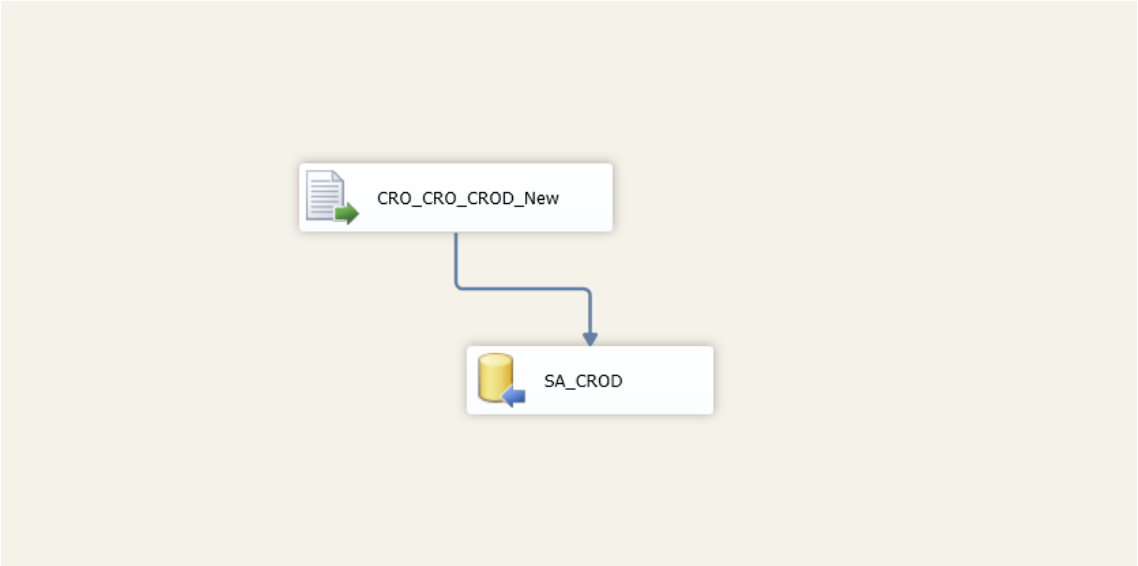

Figure 2.21: Load of Fact-Transactions SA

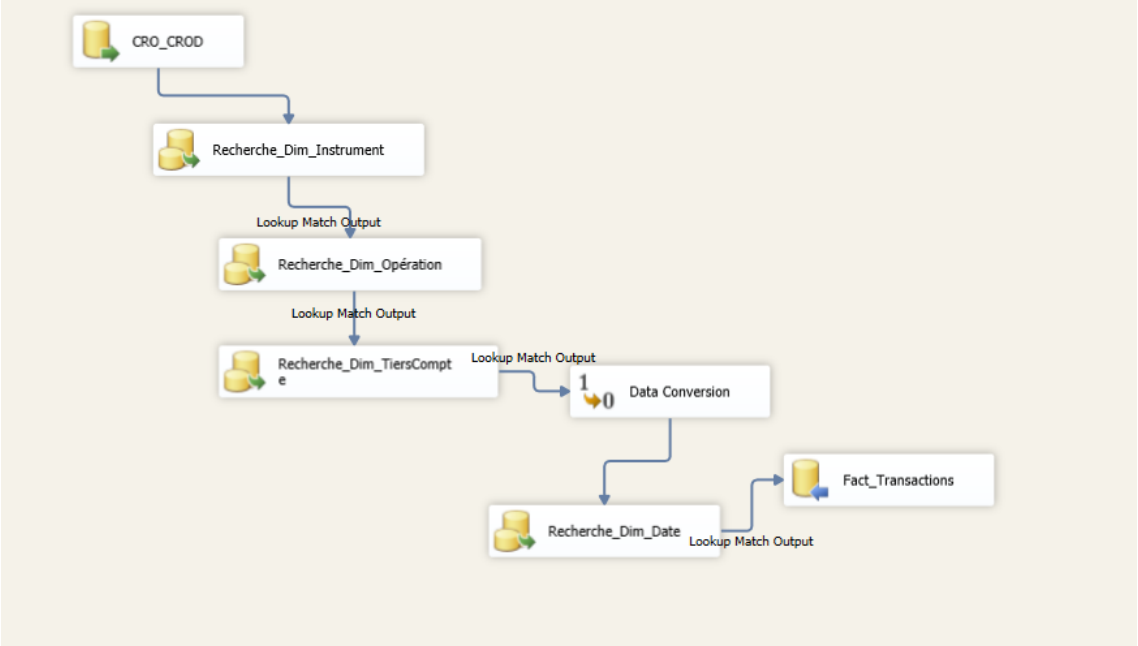

Figure 2.22: Load of Fact-Transactions DW

Chapitre 3

Data Analysis and Visualization

Plan

Introduction

In this chapter, we will move to the process of project implementation by shedding light on the various tools and technologies provided to ensure the accomplishment of this project.

3.1 Data analysis

Data analysis is a process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, informing conclusions, and supporting decision making.

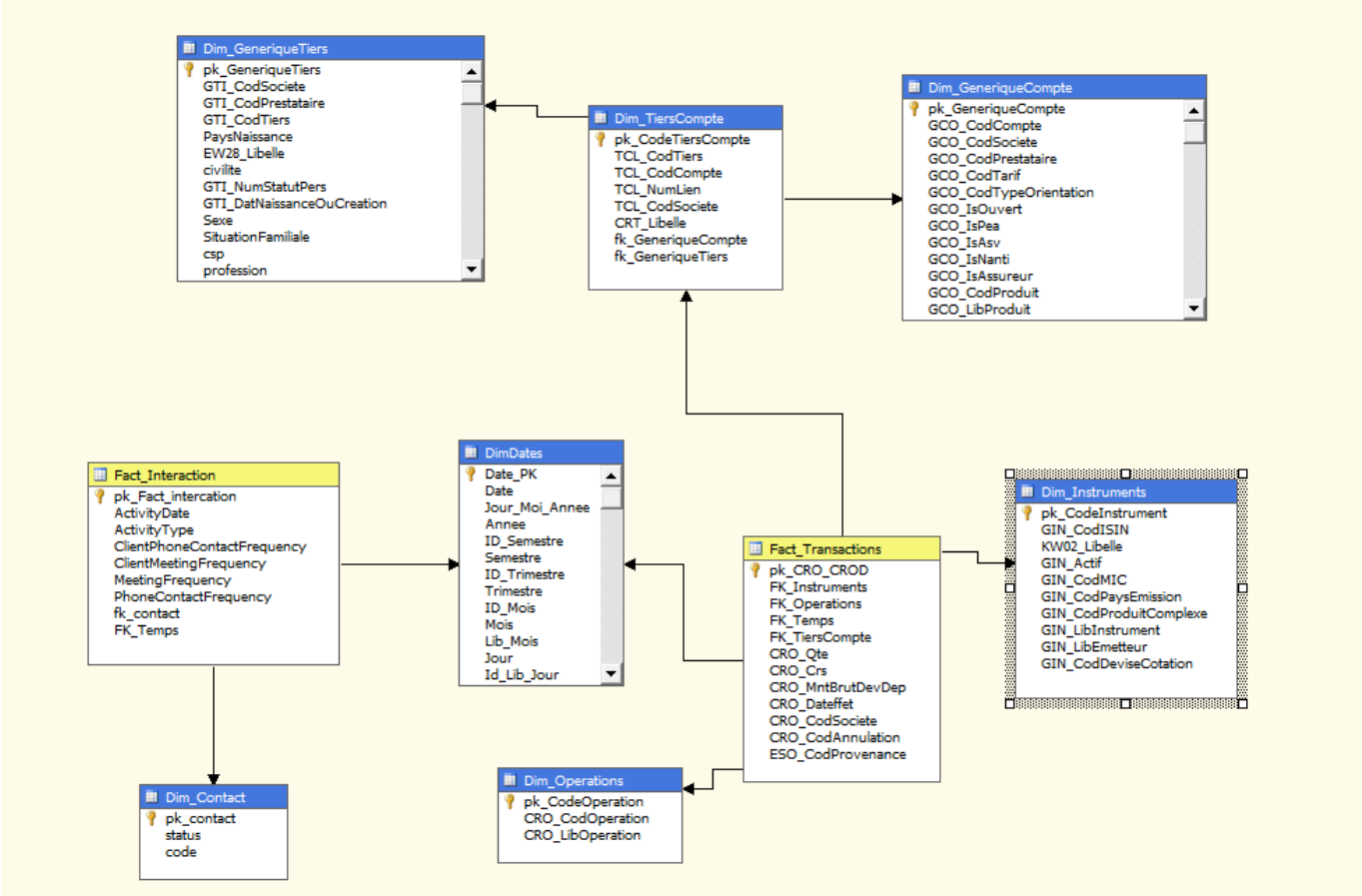

Figure 3.1: Securities Cube

3.2 Data Visualization

Reporting is a tool that should allow ODDO-BHF's managers to have a global view of their bank-customer relationship at any specific time. The Dashboard is a decision support tool. It measures performance in order to better assess how far we have come and how far we can go to access performance objectives

3.2.1 Tools and technologies

Power BI

Microsoft Power BI is a business intelligence platform that provides nontechnical business users with tools for aggregating, analyzing, visualizing and sharing data. Power BI's user interface is intuitive for users familiar with Excel and its deep integration with other Microsoft products makes it a very versatile self-service tool that requires little upfront training.

Figure 3.2: PowerBi logo

3.2.2 Dashboards

Home Dashboard

The Home dashboard contains :

- The ODDO-BHF logo.
- A button that redirects to the Admin dashboards.
- Another button that redirects to the Clients dashboard.

Figure 3.3: Home Dashboard

Admin Dashboards

The first Admin dashboard contains :

- The turnover of all the transactions done by ODDO-BHF in the selected month/year (negative is selling, positive is buying).
- Top 10 clients with the most transactions in the selected month/year.
- Top 5 best selling/buying actions in the selected month/year sorted by action values.
- Percentage of action selling/buying in the selected month/year.
- Another button that redirects to the second page of the Admin dashboard.
- Another button that redirects to the Home Page.

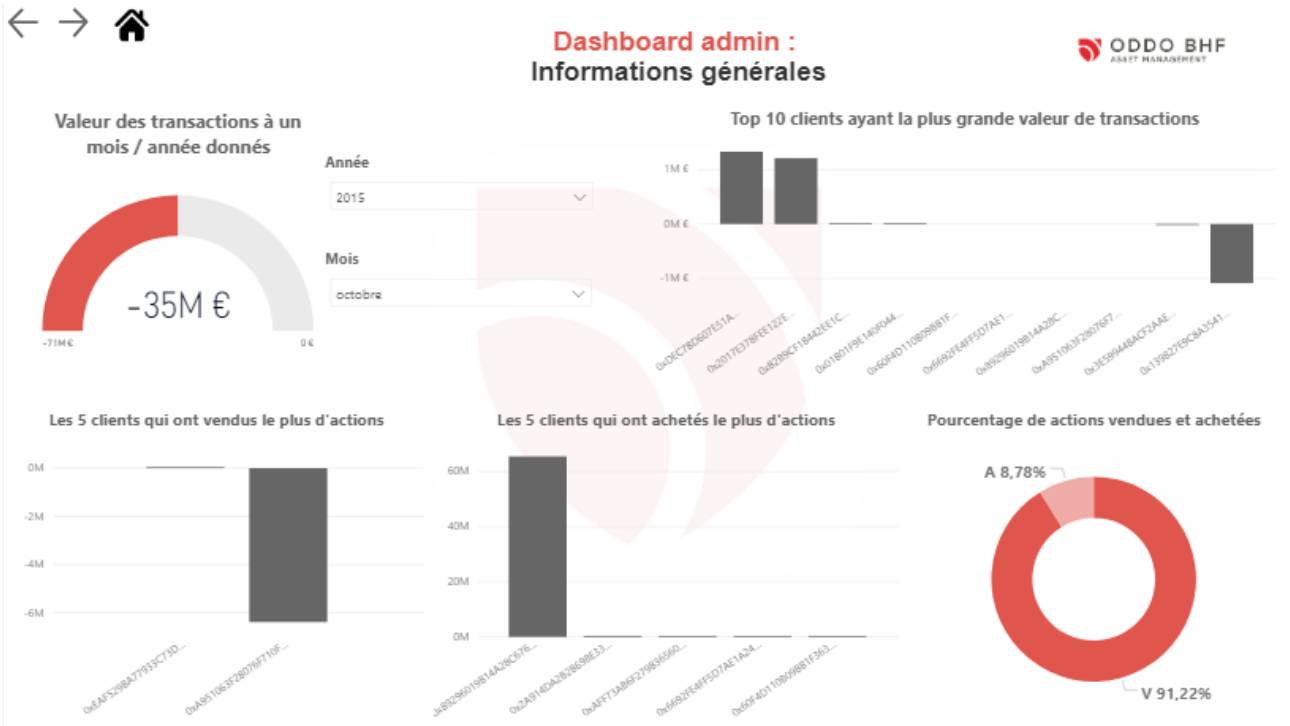

Figure 3.4: The first Admin Dashboard

The second Admin dashboard contains :

- Sum of sold/bought actions.
- Evolution of the value of a selected action over time.
- The average price of a selected action.
- Top 5 best selling/buying actions in the selected month/year sorted by action quantities.
- Top 5 worst selling/buying actions in the selected month/year sorted by action quantities.
- Another button that redirects to the third page of the Admin dashboard
- Another button that redirects to the first page of the Admin dashboard
- Another button that redirects to the Home Page

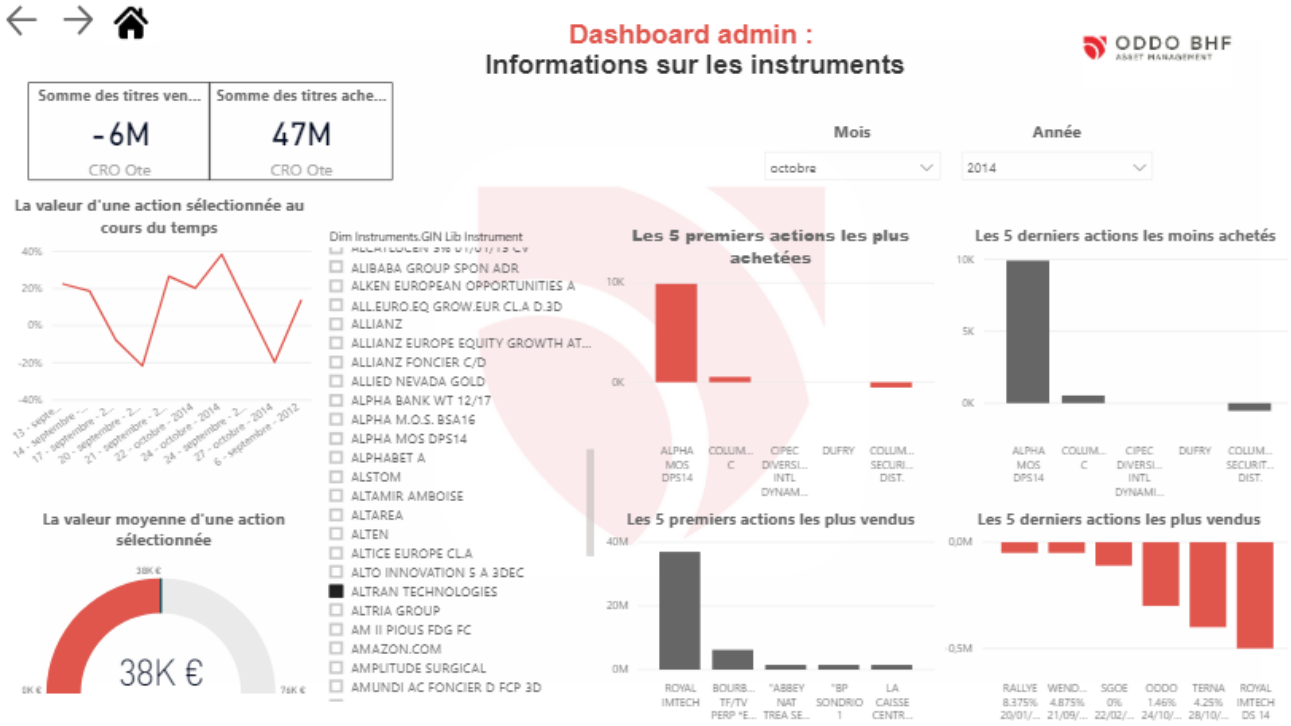

Figure 3.5: The second Admin dashboard

The Third Admin dashboard contains :

- The evolution of the phone contacts frequency over time.
- the sum of all the phone contacts.
- The evolution of the meetings frequency over time .
- the sum of all the meetings.
- Another button that redirects to the second page of the Admin dashboard.
- Another button that redirects to the first page of the Client dashboard.
- Another button that redirects to the Home Page.

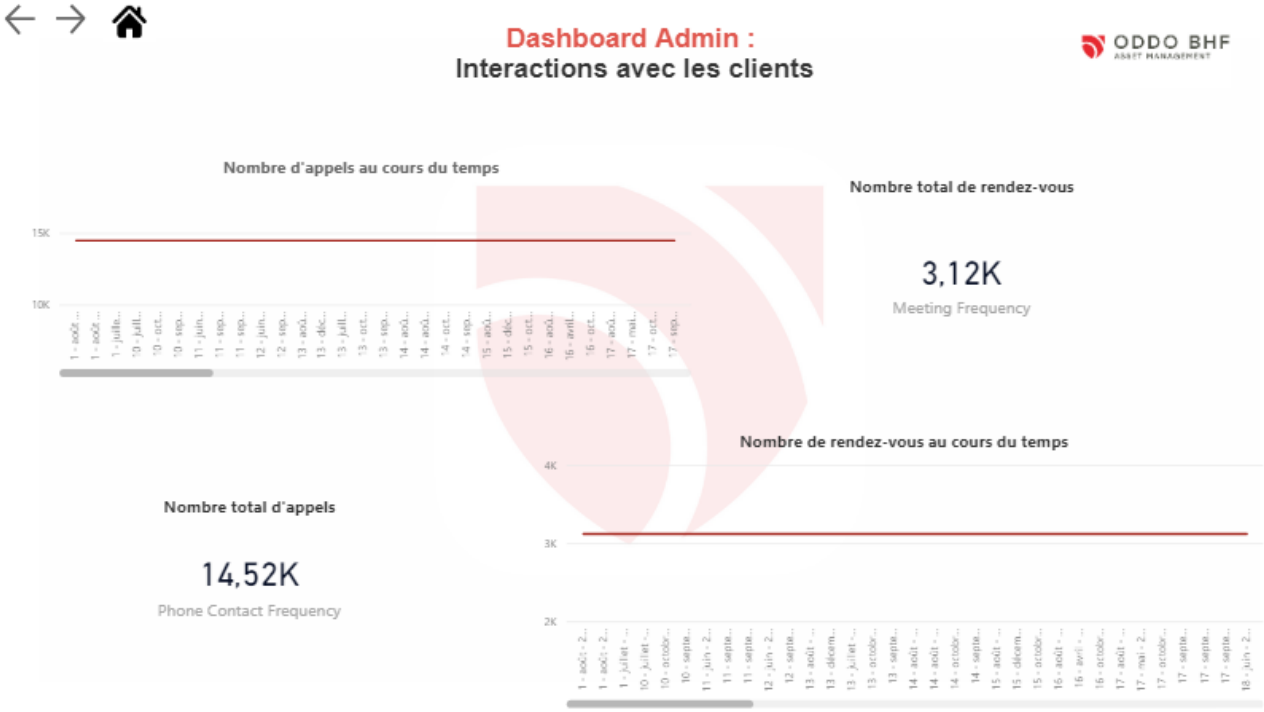

Figure 3.6: The third admin Dashboard

Client Dashboards

The first client's dashboard contains :

- A graph detailing the selected account's turnover from his operations.
- The percentage of sold/bought actions of the selected account.
- The evolution of the values of action's sum over time for the selected account.
- The evolution of the quantities of action's sum over time for the selected account.
- the sum of the phone contacts of the selected account.
- the sum of the meetings of the selected account.
- Another button that redirects to the second page of the Client dashboard.
- Another button that redirects to the Home Page.

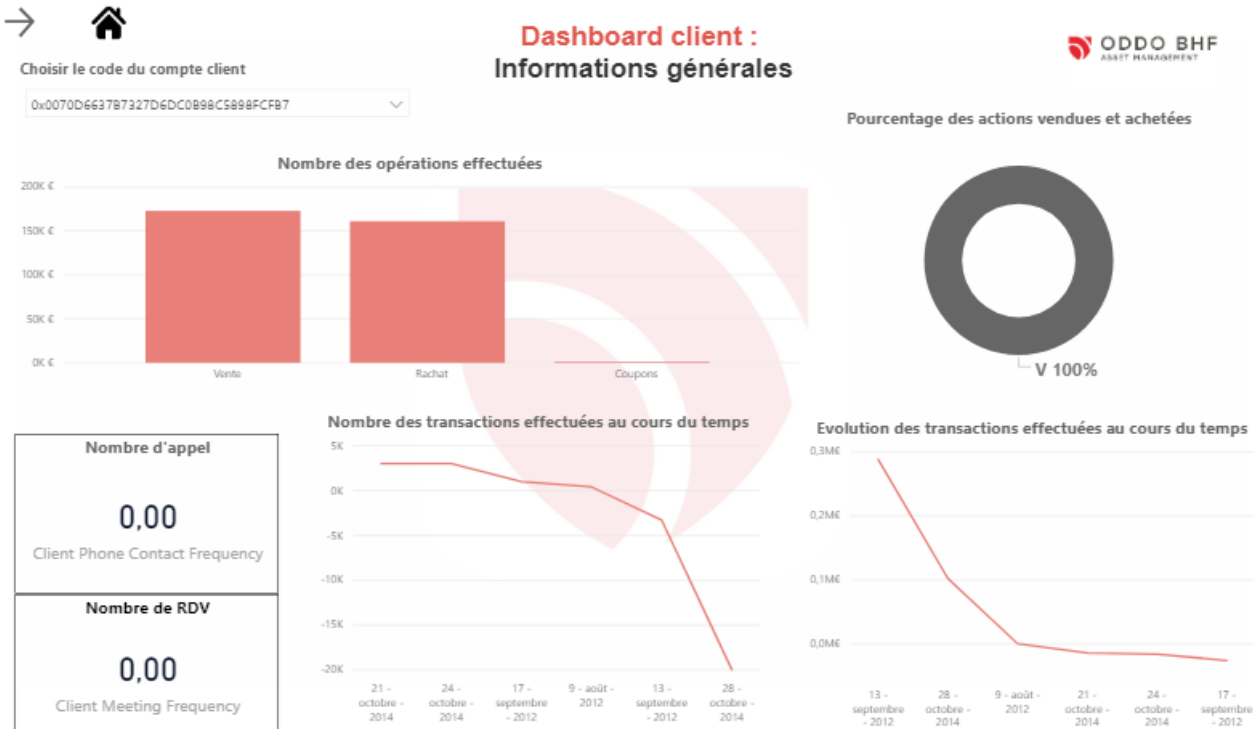

Figure 3.7: the first client Dashboard

The second client's dashboard contains :

- Sum of sold/bought actions of the selected account.
- Top 5 best selling/buying actions in the selected month/year sorted by action quantities.
- Evolution of the value of a selected action over time.
- The average price of a selected action.
- a table that contains all the transactions history of a selected client.
- Another button that redirects to the first page of the Client dashboard.
- Another button that redirects to the Home Page.

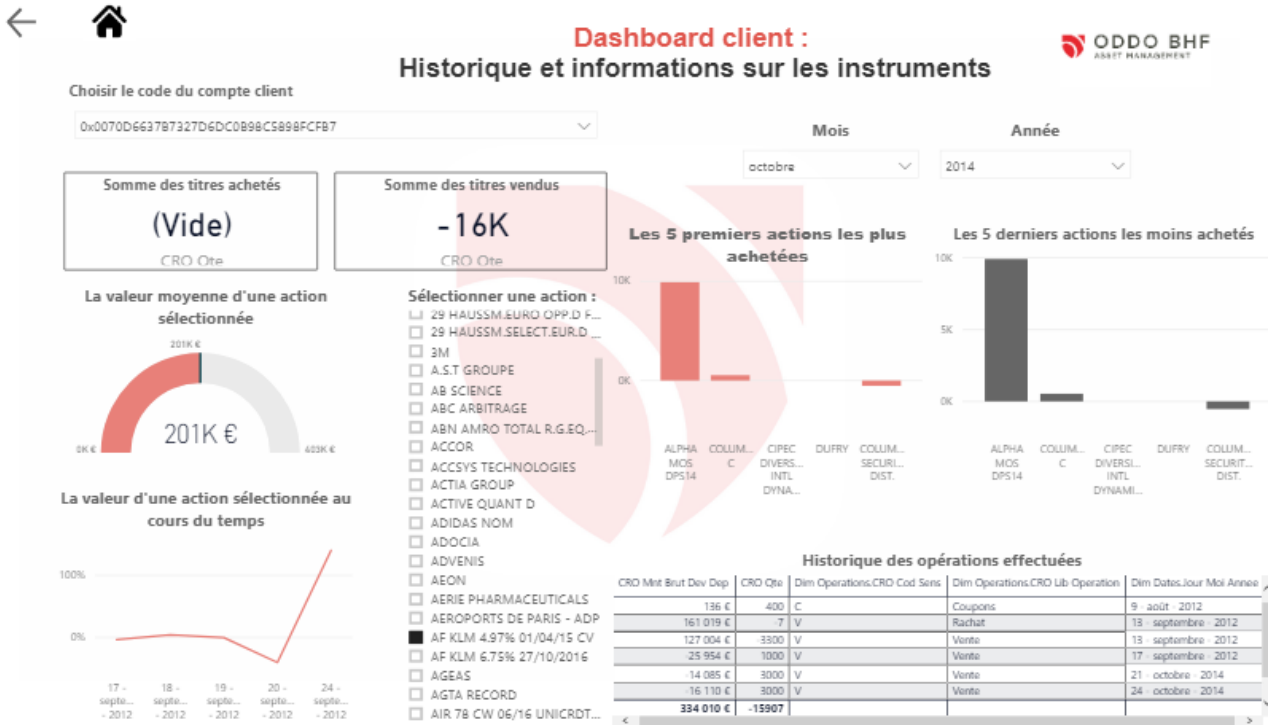

Figure 3.8: The second client Dashboard

Phone Deployment prototype

We tried deploying our dashboards on a mobile platform and this a sample of what we made.

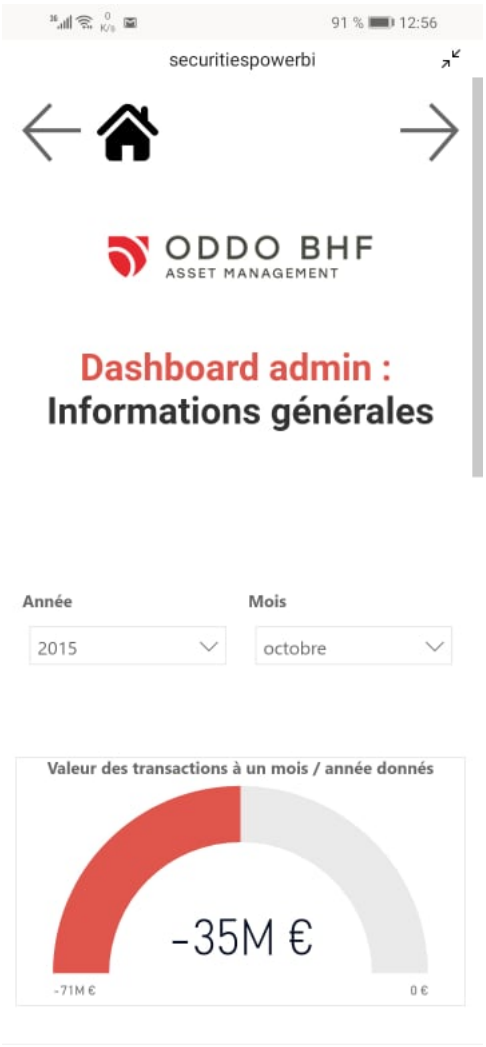

Figure 3.9: Phone deployment 1

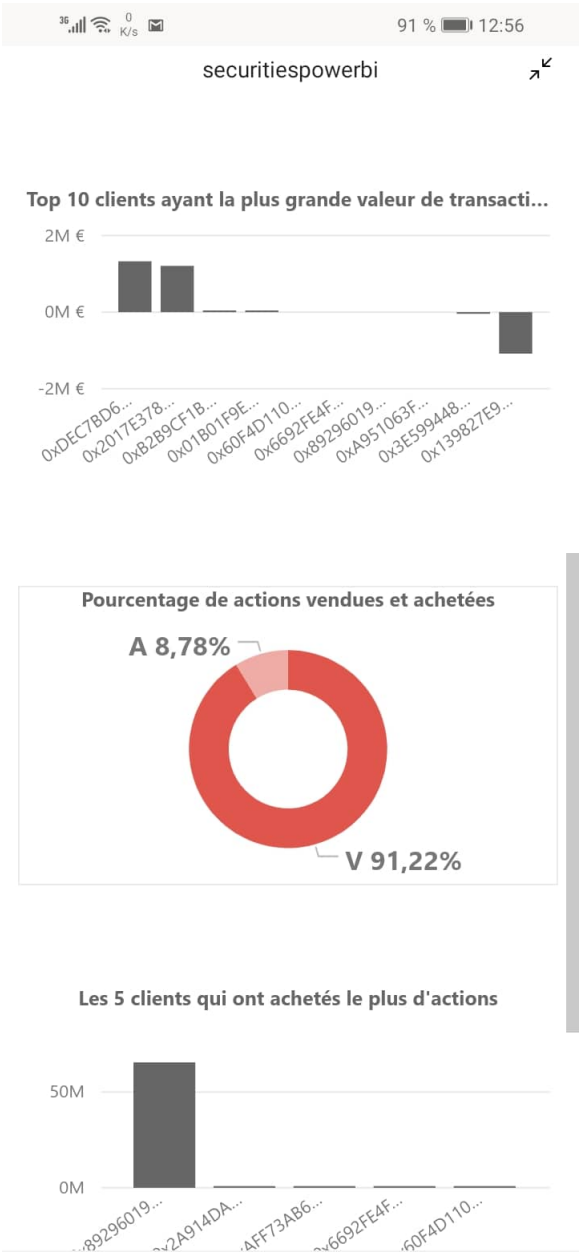

Figure 3.10: Phone deployment 2

3.3 Conclusion

In this third chapter, we analyzed and refined the provided data and our external data in order to produce a visible and simple interface to help ODDO-BHF managers choose better future course of actions to maximize customer's satisfaction and increase the turnover.

Chapitre 4

Modeling and Evaluation

Plan

4.1 Modeling

4.1.1 Profiling

K-means

We used the k-means clustering aims to partition 60660 observations into 2 clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

Kmeans

```
[16]: from sklearn.cluster import KMeans
      from sklearn.metrics.cluster import adjusted_rand_score

[17]: L = []
      for i in range(1,6):
          model = KMeans(n_clusters=i)
          model.fit(dataSansCible)
          L.append(model.inertia_)
      plt.plot(range(1,6),L)

[17]: [<matplotlib.lines.Line2D at 0x16d5426f780>]
```

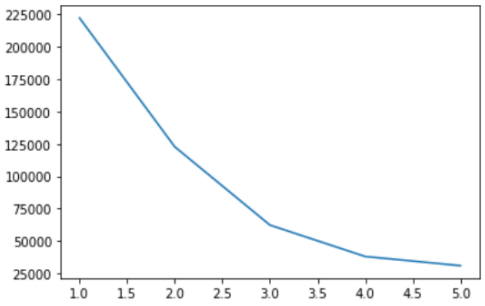

Figure 4.1: The K-means Algorithm

Evaluation

From the Cross-table, we can note that the cluster "Non risqué" have the majority of it observations in the cluster (1) and the majority of the observations of the cluster "non risqué" are in the cluster (0) .

From this model, we can deduce the characteristics of each cluster :

— Risky account

— non-Risky account

```
[24]: # Nombre de cluster = 2
kmeans = KMeans(n_clusters=2, precompute_distances='auto')
kmeans.fit(dataSansCible)
y_kmeans = kmeans.fit_predict(dataSansCible)
```

•••

```
[21]: idk = np.argsort(kmeans.labels_)
kmeans.labels_
pd.crosstab(dataCible,kmeans.labels_)
    #VP  #FP
    #FN  #VN
```

[21]:

col_0	0	1
GCO_CodToleranceRisqueMif2		
Non-Risqué	50668	3694
Risqué	68	6230

Figure 4.2: The K-means Algorithm

4.1.2 Prediction

```
[109]: sns.countplot(GeneriqueComptesNew['GCO_CodToleranceRisqueMif2'],label="Count")
plt.show()
```

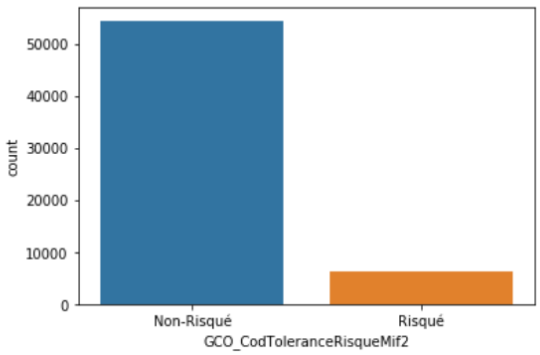

Figure 4.3: The target data verification

in this part, we will use 2 types of predictions Models that will help us to understand our new clients by predicting if they are a risked clients or not.

K-Nearest Neighbor KNN

The K - Nearest Neighbor Algorithm is a technique to classify data into a known group which is essentially predicting a specific output value using Supervised learning.

[66]:
```
from sklearn.neighbors import KNeighborsClassifier
error = []
# Calculer l'erreur pour k entre 1 et 40
#Pour chaque itération, l'erreur moyenne pour les valeurs prédites
#de l'ensemble de test est calculée et sauvegardée ds la liste Erreur.
for i in range(1, 40):
    knn = KNeighborsClassifier(i)
    knn_model = knn.fit(X_train, y_train)
    pred_i = knn_model.predict(X_test)
    error.append(np.mean(pred_i != y_test))
plt.figure(figsize=(12, 6))
plt.plot(range(1, 40), error, color='red', linestyle='dashed', marker='o',
         markerfacecolor='blue', markersize=10)
plt.title('Taux Erreur pour les differentes valeurs de k')
plt.xlabel('K ')
plt.ylabel('Erreur')
```

[66]:
```
Text(0, 0.5, 'Erreur')
```

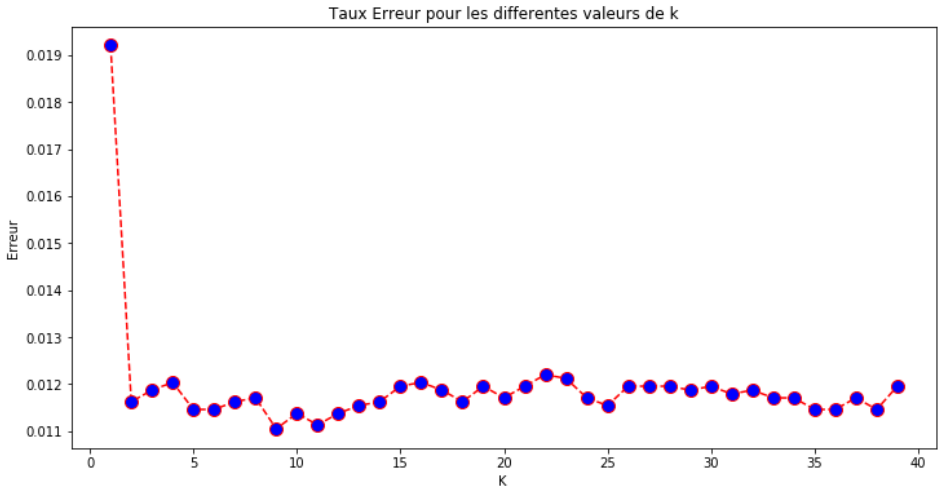

Figure 4.4: The KNN number of neighbors selection

[67]:
```
knn = KNeighborsClassifier(2)
knn_model = knn.fit(X_train, y_train)
y_pred_knn = knn_model.predict(X_test)
```

[68]:
```
print('Accuracy of K-NN classifier on training set: {:.2f}'
      .format(knn.score(X_train, y_train)))
print('Accuracy of K-NN classifier on test set: {:.2f}'
     .format(knn.score(X_test, y_test)))
```

```
Accuracy of K-NN classifier on training set: 0.99
Accuracy of K-NN classifier on test set: 0.99
```

[69]:
```
from sklearn.metrics import confusion_matrix
print(confusion_matrix(y_test, y_pred_knn))
```

```
[[10839    18]
 [  123  1152]]
```

Figure 4.5: The KNN Algorithm

Random-Forest

A Random Forest consists of a collection or ensemble of simple tree predictors, each capable of producing a response when presented with a set of predictor values.

Random Forest ¶

```
[74]: #random forest kima arbre decisionnel ama tkharajlek ahsen arbre fil
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier()
rfc_model = rfc.fit(X_train, y_train)
y_pred_rfc = rfc_model.predict(X_test)
```

```
C:\Users\iheb_\Anaconda3\lib\site-packages\sklearn\ensemble\forest.py:245: FutureWarning: The default value of n_estimators will
change from 10 in version 0.20 to 100 in 0.22.
  "10 in version 0.20 to 100 in 0.22.", FutureWarning)
```

```
[75]: print('Accuracy of Random Forest classifier on training set: {:.2f}'
     .format(rfc.score(X_train, y_train)))
print('Accuracy of Random Forest classifier on test set: {:.2f}'
     .format(rfc.score(X_test, y_test)))
```

```
Accuracy of Random Forest classifier on training set: 0.99
Accuracy of Random Forest classifier on test set: 0.99
```

Figure 4.6: The Random forest algorithm

```
[76]: from sklearn.metrics import confusion_matrix
print(confusion_matrix(y_test, y_pred_rfc))
```

```
[[10815    42]
 [   91  1184]]
```

```
[77]: from sklearn.metrics import classification_report
print(classification_report(y_test, y_pred_rfc))
```

```
              precision    recall  f1-score   support

           0       0.99      1.00      0.99     10857
           1       0.97      0.93      0.95      1275

    accuracy                           0.99     12132
   macro avg       0.98      0.96      0.97     12132
weighted avg       0.99      0.99      0.99     12132
```

Figure 4.7: The Random forest results

Evaluation

```
[97]: models = pd.DataFrame({
          'Model': ['KNN',
                    'Random Forest',
                    'Support Vector Machine'],
          'Score': [acc_knn, acc_rfc,
                    acc_svm, ]})
      models.sort_values(by="Score",ascending=False)
```

[97]:

	Model	Score
1	Random Forest	0.989037
2	Support Vector Machine	0.988460
0	KNN	0.988378

Figure 4.8: The comaratif table

ROC Curve

```
[87]: plt.figure()
      plt.plot(fpr1, tpr1, color='blue', lw=2, label='Support Vector Machine (area = %0.2f)'% roc_auc1)
      plt.plot(fpr2, tpr2, color='green', lw=2, label='Random Forest ROC curve (area = %0.2f)'% roc_auc2)
      plt.plot(fpr3, tpr3, color='yellow', lw=2, label='kNN ROC curve (area = %0.2f)'% roc_auc3)
      plt.plot([0, 1], [0, 1], color='red', lw=2, linestyle='--')
      plt.xlim([0.0, 1.0])
      plt.ylim([0.0, 1.05])
      plt.xlabel('False Positive Rate')
      plt.ylabel('True Positive Rate')
      plt.title('Classifiers ROC curves')
      plt.legend(loc = "lower right")
      plt.show()
```

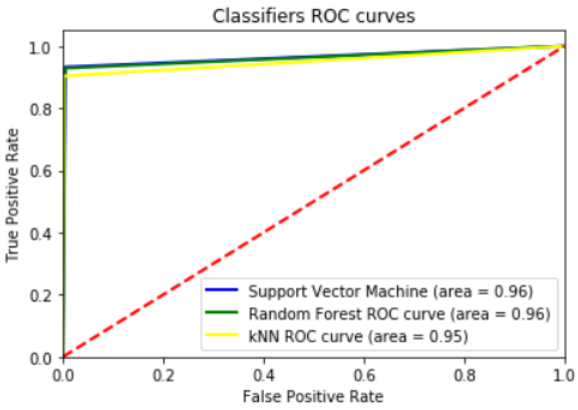

Figure 4.9: The ROC Curve

Conclusion

In this final chapter, we used algorithms elaborated by Python in order to have a better vision of our new clients and predict their efficiency against the ODDO-BHF company.

General Conclusion

Our team presented through this report all of the phases of our Business intelligence project during which we set up our objectives, refined and integrated our data and finally provided a thoughtful dashboard that enables the Oddo bhf group to better communicate with its clients and insure the best service quality by determining the best plans and move through our data analysis for greater investments and better revenues.

This project was an opportunity for us to learn new technologies and to deal with problems in a short amount of time, but the most important thing is the acquisition of teamwork skills.