University of Strasbourg

Master CSMI

Intership Report : Business intelligence

Authors:
Melissa Aydogdu

Supervisors:
Christophe Prudhomme
Alexis De Rivoyre

Date: August 20, 2023

Contents

0.1 Acknowledgements

I would like to express my gratitude to Mr DE RIVOYRE, my internship supervisor, for his kind guidance during my internship. His enlightened advice, pertinent guidance and constant availability were essential to my professional development.

I would also like to express my gratitude to my colleagues from Atos, especially KENNE NOTEMI Leslie Stella and NOUMI Yves Joël, who taught and helped me during my internship; their expertise guided me throughout this enriching experience.

Finally, I also want to extend my gratitude to my academic supervisor and my Master supervisor Mr PRUDHOMME, as well as to all my teachers who have supported me throughout my studies. Your support and teaching have played a decisive role in my education and my academic and personal development.

0.2 Introduction

0.2.1 Digital Service Companies (DSC)

Formerly known as "Computer Engineering Services Company" (SSII or SS2I), an IT consulting company is a firm that provides specialized expertise, guidance, and services related to technology and information systems to other businesses or organizations. These companies offer their knowledge and skills to help clients effectively leverage technology to achieve their business objectives, improve efficiency, and overcome technology-related challenges.
The services offered by IT consulting companies can vary widely based on their areas of expertise and the specific needs of their clients..
Five major companies share the podium of leaders in the digital services sector including:

- CAPGEMINI
- IBM
- Orange Business Services (OBS)
- ATOS
- ACCENTURE

The next paragraphs will be dedicated to the presentation of the company ATOS.

0.2.2 Presentation of Atos

The project was submitted and managed by Atos. Atos is an international leader in digital transformation with 112,000 employees and annual revenues of approximately 11 billion euros. The Group is the European leader in cloud computing, cybersecurity and supercomputing, providing integrated solutions to all sectors in 71 countries. As a pioneer of decarbonization services and products, Atos is committed to delivering secure, decarbonized digital solutions to its customers. Atos is an SE (European Company) listed on Euronext Paris.
Atos' purpose is to help shape the information space. With its skills and services, the Group supports the development of knowledge, education and research in a multicultural approach and contributes to the development of scientific and technological excellence. Everywhere in the world, Atos enables its customers and employees, and more generally the general public, to live, work and progress sustainably and with confidence in the information space.
They develop and produce solutions in the following sectors :

- Financial services and insurance;
- Health and Life Sciences;
- Manufacturing industry;

- Public Sector and Defense;
- Resources and Services;
- Telecommunications and Media;

Atos works closely with technology leaders to accelerate the creation of value for its customers and provides world-class solutions for all businesses.

Figure 0.2.1: Collaborator of Atos

For more informations, refer to the Atos website.

0.2.3 Business Intelligence

The term Business Intelligence (BI) refers to the technologies, applications and practices for collecting, integrating, analyzing and presenting information. The objective of Business Intelligence is to support better decision making in the business, sales, marketing and finance verticals. Essentially, Business Intelligence systems are data-driven decision support systems. Business Intelligence is the technological process of analyzing data and presenting information to help executives, managers, and other end-users in the company make informed business decisions. Business Intelligence encompasses a wide variety of tools, applications and methodologies that enable organizations to collect data from internal systems and external sources. This data is then prepared for analysis to create reports, dashboards and other data viz tools to make analytical results available to decision makers and operations.
Today, companies rely on business intelligence software to identify and extract valuable information from the large volumes of data they store. These tools can be used to extract information such as competitive intelligence and market trends, as well as internal information such as finding the reasons for lost opportunities.
The job of a business intelligence engineer consists of managing large volumes of data within companies in order to help them improve their performance. The BI engineer must:

- Analyze data within a limited timeframe defined upstream. This data comes from the company's various IT systems but can also come from outside;
- Produce reports in order to offer a global view to the business departments;
- Design and model the databases needed to produce the reports;
- Test reporting tools and dashboards;
- Integrate its developments;
- Maintain and develop the system.

0.2.4 The main stages of a decision support project

The implementation and success of a BI project involves several steps:

Collection of the need and formalization of the project

This phase is one of the most important of the project because it allows us to understand the needs and expectations of the client. The collection of these needs is generally done through three phases of meetings between the client and the project manager:

- The kick-off meeting: the purpose of this meeting is to present the project's objectives to all the stakeholders, to remind and validate the project's scope, to present the implementation approach developed by the project manager in order to best meet the objectives raised by the client and finally to present the project's provisional schedule.
- The analysis and needs gathering workshops: they allow to gather and formalize the business needs of the users in the form of several deliverables to be produced, while ensuring that these needs are understood and shared by all stakeholders. The deliverables to be produced following this phase are: the needs maps, the data dictionary and the preliminary analysis report .
- Mock-up workshops: This stage takes place when the client raises the need for dashboards to the project manager. The mock-ups allow the client to project the form that will be used to present the synthesized data to users. This ensures that the data will be rendered in a way that is understandable for the users. It should be noted that the client can produce a graphic charter that will be used for all its dashboards.

The phase of collecting the needs and formalizing the project also allows us to identify all the data sources necessary for the exploitation of the data of the studied sector of activity.

Modeling the data warehouse

After the first stage has been produced, the next step is to translate the business requirements into detailed functional and technical specifications that will enable the data warehouse to be modeled. The data warehouse plays an essential role in the BI information system, as it allows data to be stored in an integrated, subject-oriented, non-volatile, historical, summarized structure that is available for query, analysis and retrieval. In the modeling process it is necessary to :

1. Create the fact tables: these contain the evaluation and decision support tools that have been requested by the client. These tools are still called "indicators" in the decision-making jargon.

2. Create the different dimensions: these are the analysis axes that will be used to process the data stored in the fact tables.

Data supply

Once the structure of the warehouse has been completed, we move on to the implementation of the warehouse data supply. The purpose of this feeding is to make the link between the data sources and the data warehouse in an automatic and planned way. This is done using ETL (Extract Transform Load) software. It allows the three (3) main steps of data feeding within a warehouse to be carried out, namely

1. Extraction (Extract): collection of data from the various sources available (databases, flat files, excel, csv etc.)

2. Data transformation (Transform): application of calculation rules and transformations to the data (transmitted by the client) and adaptation of the data to the warehouse structure.

3. Loading: loading the data into the final warehouse tables.

Business layer modeling

Once the data warehouse has been loaded, it is now necessary to define a business data layer that allows the data to be manipulated with a vocabulary that can be understood by the restitution tools and users. A central element of the system, it enables BI applications to mask the diversity of data origin and to benefit from a common, homogenous, standardized and reliable source of information within a system.

Data restitution

Data restitution, also called reporting, consists in presenting the data processed and stored in the data warehouse in a synthetic or detailed way in the form of interactive dashboards and graphs. It is important to take care of the data restitution because it is the part of the BI system that will be the most visible for the end user.

0.3 TPAM BI

0.3.1 Goal of the intership

Immersion in a production context dedicated to the implementation of business intelligence solutions for one of Atos' clients.

- Modeling and detailed design from the expression of customer needs and functional specifications.
- Development of data integration flows and development of dashboards and steering. Data quality management.
- Unit and integration tests.
- Support for acceptance and corrective actions.
- Technology watch.

Skills:

- To develop your methodological and technical knowledge on business intelligence projects.
- A functional base in the sectors of communities, administration, health, industry.
- To propose innovative solutions to the needs of an application team.

0.3.2 TPAM

Third Party Application Maintenance ("TPAM") is when a company transfers all or part of the maintenance of its applications to a third party IT service supplier, in order to ensure their proper operation.
An increasing number of companies are choosing to externalize the management of their information systems (IS) within the framework of TPAM, which is generally accompanied by a long-term contract. For the service provider, the aim is to ensure that the application in question is running smoothly, by correcting any faults identified by users and carrying out any necessary updates. A distinction must be made between third party application maintenance and evolutionary maintenance, the latest of which consists in adding new functionalities to an application.

Within the agency Atos, the employees are divided into several teams, each team works for different client, for my internship I was brought to integrate the team of TPAM which works for the ES Energie. This team is composed of 2 different parts, a Java part, which takes care of the different applications, and the other BI team which takes care of the business intelligence part.

Électricité de Strasbourg (ÉS) is a French public limited company that distributes and markets electricity and gas in the Bas-Rhin region. Founded in 1899, the Strasbourg-based company is now part of the Électricité de France (EDF) group. The group is considered a major contributor to Atos in the Grand Est region.

0.3.3 Operating mode

The TPAM team works in the form of a ticket which is received via an application called picto. Each team (BI/JAVA) receives requests from the ES customer as tickets for change requests, service requests or incident alerts of various priorities (P1, P2, P3).

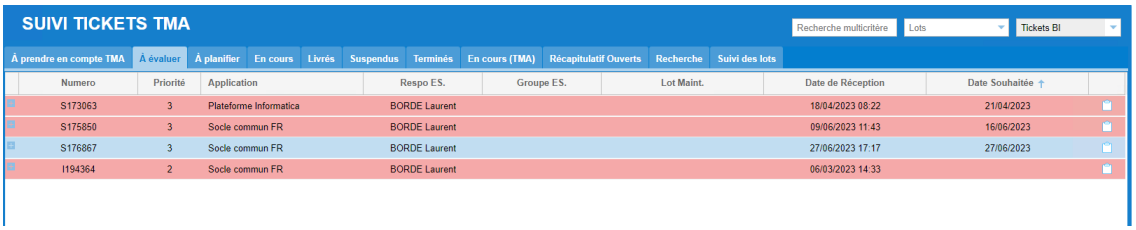


Figure 0.3.1: Ticket monitoring

These tickets must be processed within a certain time limit in order to meet our commitments to the customer. The following table summarizes these commitments:

breakdown service commitment	Type of request	Processing times	Maximum lead times
Priority 1 " blocking "	Incident	4 hours on working days	1.5 working days
	Service	4 hours on working days	1.5 working days
	Change	2 days	5 days
Priority 2 " major "	Incident	Time limit fixed by the parties	Time limit fixed by the parties
	Service	Time limit fixed by the parties	Time limit fixed by the parties
	Change	Time limit fixed by the parties	Time limit fixed by the parties
Priority 3 " minors "	Incident	Time limit fixed by the parties	Time limit fixed by the parties
	Service	Time limit fixed by the parties	Time limit fixed by the parties
	Change	Time limit fixed by the parties	Time limit fixed by the parties

Figure 0.3.2: Contractual commitment

0.3.4 Ticket delivery process

For every processing or request there is a well-defined procedure for handling them. The first step is to analyze and evaluate the request in order to estimate the cost of the ticket, indicating the number of days required for realization. Then we can start to process the request in a development environment, all of the tools that I'm going to present in the following sections have 3 different

environments. First of all, we have the development environment, which enables us to execute the processes and test them with examples. In this environment, we can, for example, modify or add a field to a table in order to test whether our processes are working correctly. Once we've finished developing and testing, we proceed with a first delivery to the test environment. The customer can then check the process in the test to give us the green light for delivery to production.

One of the tools we use for delivery is putty, which enables us to deliver workflows (see part 0.4.3) that we export via repository manager (see part 0.4.2). Putty is a terminal client software, mainly used to establish secure connections with remote servers via the SSH (Secure Shell) protocol. It is compatible with Windows and Linux operating systems.
The aim is to deploy PWC Informatica feed processing from the development environment to the test environment, and then after validation to the production environment.
These deployments must have the following characteristics:

- Automaticity: a maximum number of operations will be encapsulated in a script.
- Reversibility: it must be possible to return to the previous situation.
- Security:
  - Only the Informatica operating account for AIX servers will be able to carry out a deployment.
  - Prohibit simultaneous deployments.
  - Guarantee platform integrity.
- Traceability: all operations will be logged in a "LOG" file.

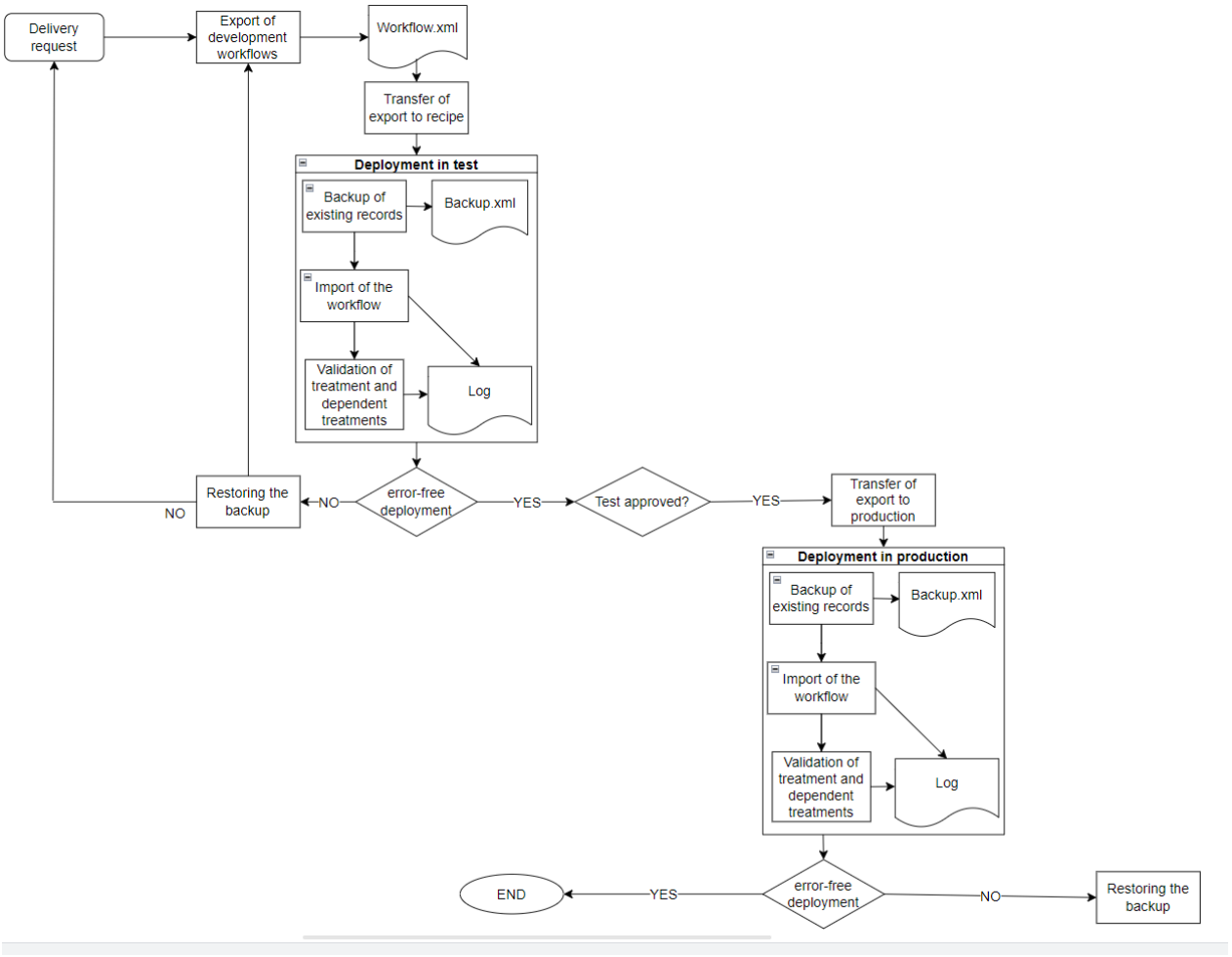


Figure 0.3.3: Informatica treatment delivery process

Here's a diagram explaining the delivery process, as explained before we create and test our treatments in the dev environment and then make a first delivery in test environment, this is done with putty by executing a script, and in order to take all the necessary precautions when executing the script on putty this takes care of copying the old version of the treatment in order to store it in a folder in case of problems. It's also possible to find a log file that summarizes the entire execution and details any errors during delivery. After validation by the customer, we'll need to follow the same steps for the delivery in production.

0.3.5 SQL DEV

SQL Developer software (Oracle SQL Developer) is a free, powerful integrated development environment (IDE), provided by Oracle Corporation, for the creation and management of Oracle databases.

It is designed to facilitate the development, testing and administration of databases, as well as the writing and execution of SQL queries.
Since we work with SQL Server databases, we use SQL Dev with a SQL Server extension.

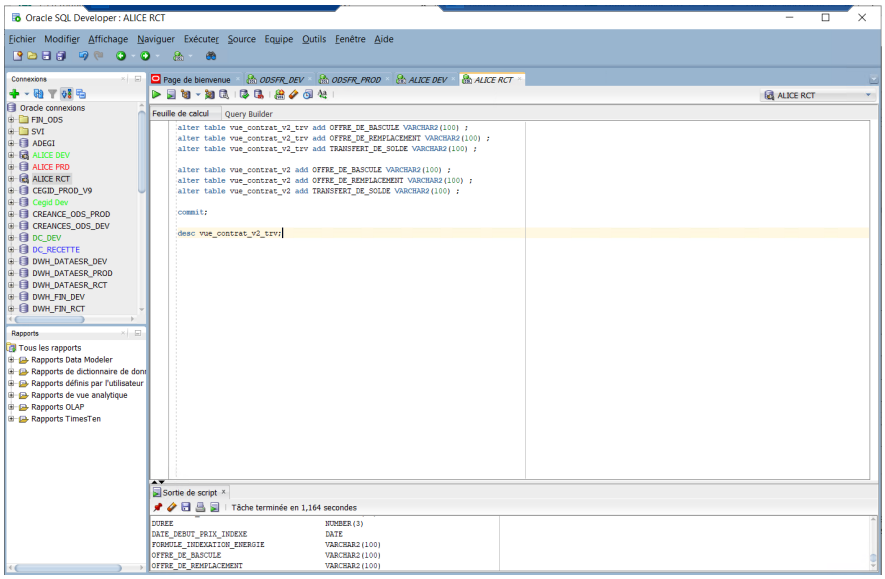


Figure 0.3.4: SQL Dev interface

0.3.6 POWER AMC

Power AMC software is a modeling and design tool for information systems. Developed by SAP, Power AMC is widely used by IT professionals to design business architectures, data models, UML (Unified Modeling Language) models and other diagrams for software design.
It allows you to design relational data models using entity-relationship (ER) diagrams, or logical and physical data models.

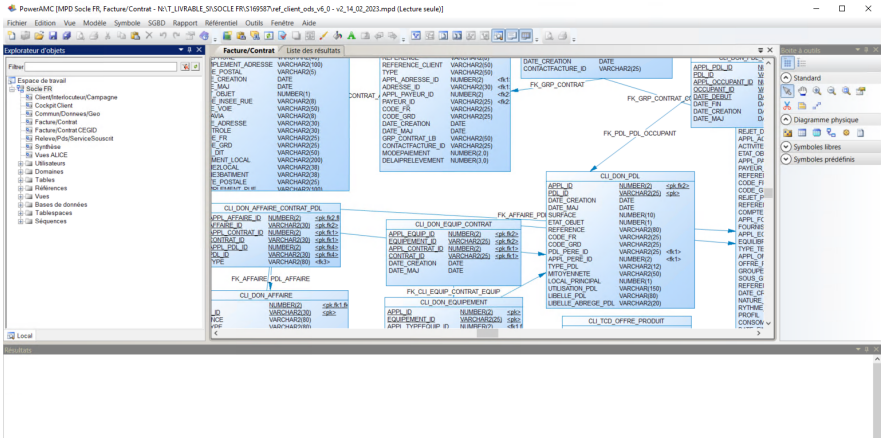


Figure 0.3.5: POWER AMC interface

0.3.7 SAP Web intelligence

SAP BI (Business Intelligence) is a software package developed by SAP, a company specializing in business solutions. SAP BI, also known as SAP BusinessObjects, is designed to help companies make smarter decisions by providing data analysis, reporting and visualization capabilitie.

One of the most frequently used elements is the SAP BusinessObjects Information Design Tool (IDT). It is used to create and manage universes, which are logical structures for accessing and querying data in a simplified way.

This tool is used to design and create universes, which are logical data models. In IDT, you can define classes, business objects, joins and conditions to structure your universe. You can also define hierarchies, value lists and functions to enrich the semantics of your universe. The program also makes it easier to manage versions and deploy universes. You can create different versions of a universe, keep track of changes and publish them in production environments. This enables efficient management of universe evolutions and updates.

The Web Intelligence tool can work as a client tool of the BusinessObjects platform and also as a stand-alone reporting tool.

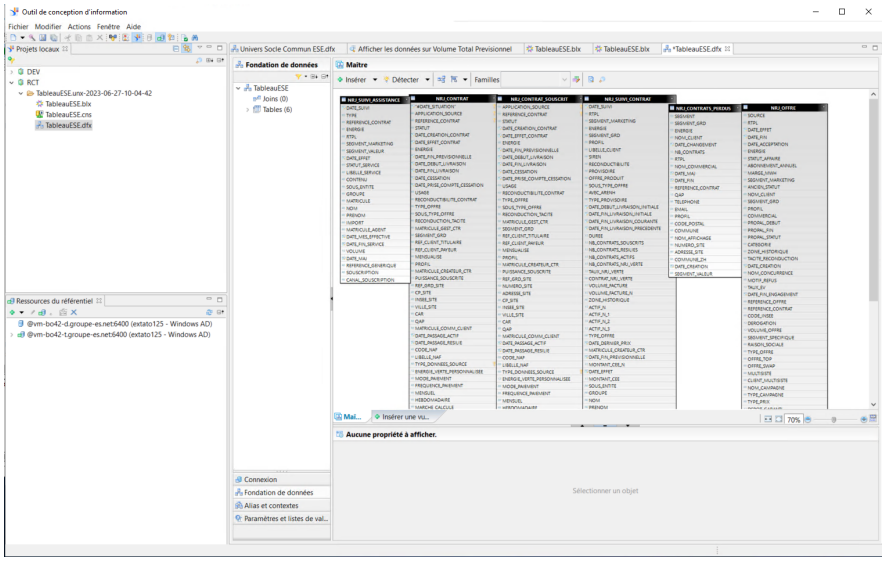


Figure 0.3.6: SAP IDT interface

Now let's try to understand what a universe is. In SAP BI, a universe is a logical layer that simplifies data access and query. It is a business data model that allows users to build queries and obtain relevant results without having to worry about the technical complexities of the associated data sources. It is a logical structure that defines how users view and interact with data and it represents business objects, such as tables, columns and relationships, that are relevant to users and for which the complexity of the data is not visible. It provides a consistent, consolidated view of the data, enabling users to formulate queries based on familiar business concepts.
On the other hand, we also have the SAP Central Management Console (CMC), which is a centralized management tool for SAP BusinessObjects applications. It provides a user-friendly web interface for managing the various components of the SAP BusinessObjects solution package, such as servers, users, access rights, schedules, publications and so on.
Integration between IDT and CMC comes into play when you want to deploy a universe created in IDT to the production environment managed by CMC.

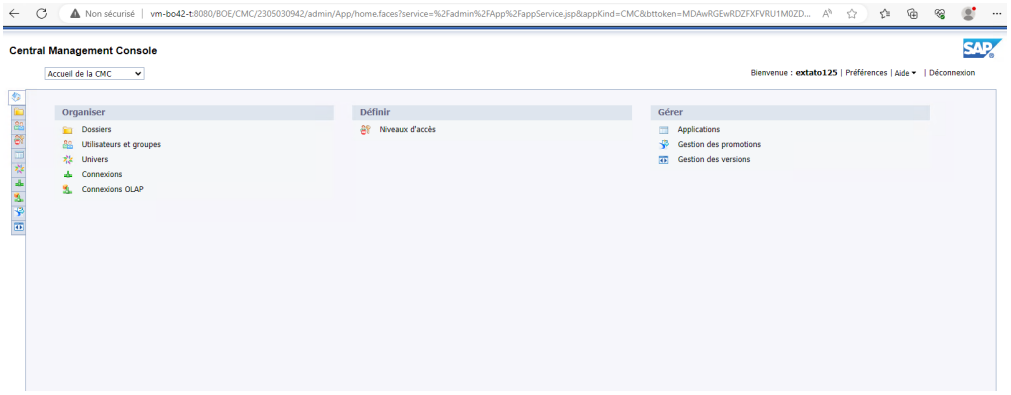


Figure 0.3.7: SAP CMC interface

The universes created in SAP IDT are then used in the interactive web-based reporting and analysis tool SAP BusinessObjects Web Intelligence (WebI). It allows users to create reports, navigate through data, perform analyses and share results with other users.
Using Web Intelligence, business users can create basic, medium and complex reports from transactional data in a database and by creating universes using the Information Design Tool/UDT. Various SAP and non-SAP data sources can be used to create reports in Web Intelligence. The SAP Business Warehouse (BW) system does not require a universe to connect to the Web Intelligence tool.

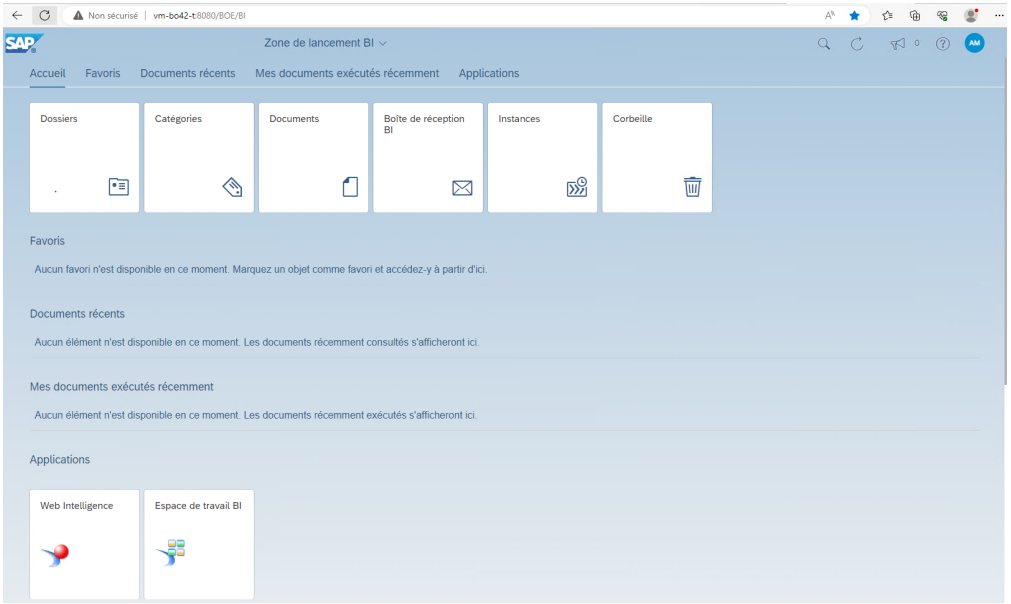


Figure 0.3.8: SAP BI interface

0.3.8 FileZilla

FileZilla is an open-source FTP (File Transfer Protocol) client software used to transfer files between a local computer and a remote server. It enables users to connect to FTP, SFTP (SSH File Transfer Protocol) and FTPS (Secure FTP) servers to upload, update and manage files on these servers. In the context of this internship, we are storing source/target files or xml files on the development, test and production servers, to enable us to deliver processing.

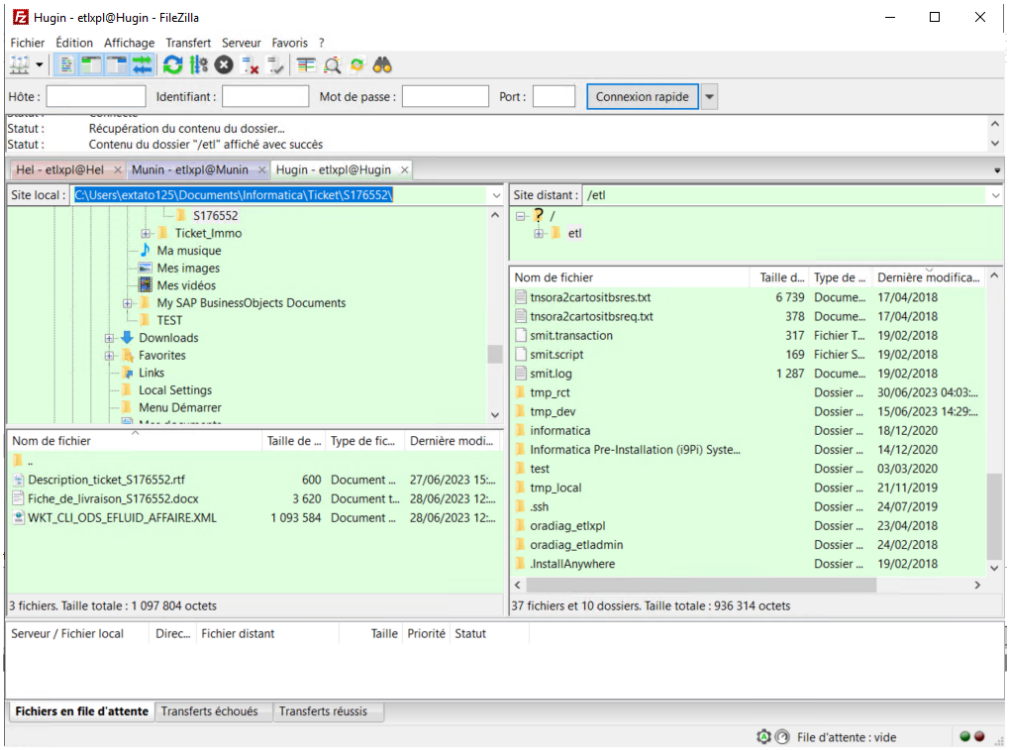


Figure 0.3.9: SAP BI interface

0.4 Informatica

0.4.1 Introduction

Informatica has several products focused on data integration, but the product we will use more often during the internship is Informatica PowerCenter, which is based on an ETL architecture. It provides data integration software and services to a variety of businesses, industries, and government organizations, including telecommunications, healthcare, financial services, and insurance. Let's first try to understand what data integration is for and in what context we can use it. Let's first try to understand what is data integration.

Data integration simply put combines data types and formats into a single location often know as a data warehouse. The ultimate goal of data integration is to gain useful and valuable information to help solve problems and gain new insights while organizations spend a lot of time and money collecting data. Now let's try to understand how to do data integration using ETL, first we have to explain what is ETL.

Etl is a process that extracts transforms and loads data from multiples sources to a data warehouse or other unified data repository. ETL stands for extract, transform, load and it is a data integration process in order to integrates data from multiple data sources into a single consistent data store that is put into warehouse or other destinations systems.

Now let's look at the steps required for the ETL process:

1. In the first step data is extracted from a source, into a stacking area, the stacking area acts as a buffer between the data warehouse and the source. Since data may be coming from multiple different sources it's likely in various formats and directly transferring the data to the warehouse may result in corrupted data. The stacking area is used for data cleansing an organization.

2. In the second step which is the transformation phase, data is processed to make its values and structure conform consistently with its intended use case. The goal of transformation is to make all data fit within a uniform schema before it moves on to the last step.

3. Finally, the load phase moves the transformed data into a permanent target system. This could be a target database, data warehouse, data store, data hub or data lake — on-premises or in the cloud. Once all the data has been loaded, the process is complete.

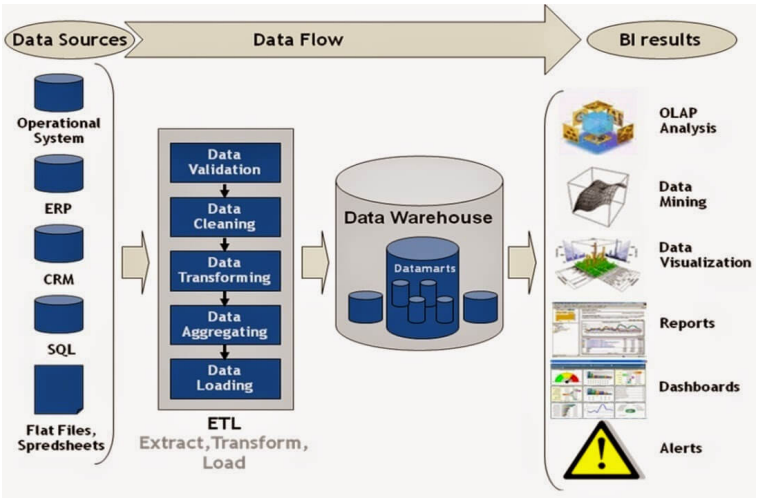


Figure 0.4.1: ETL process

0.4.2 Interfaces of informatica

PowerCenter Repository Manager

The repository manager allows you to administer repositories. Within the structure of the TMA, we are brought to make process deliveries, in these cases we use the repository manager to export objects like worflow. We can also navigate through several folders and repositories and perform the following tasks:

- Manage user and group permissions. Assign and revoke permissions for folders and global objects.
- Perform folder functions. Create, edit, copy and delete folders. Work done in the Designer and Workflow Manager is stored in folders. If you want to share metadata, you can set up a folder to be shared.
- View metadata. Analyze the sources, targets, mappings, and dependencies of shortcuts, search by keyword, and view the properties of repository objects.

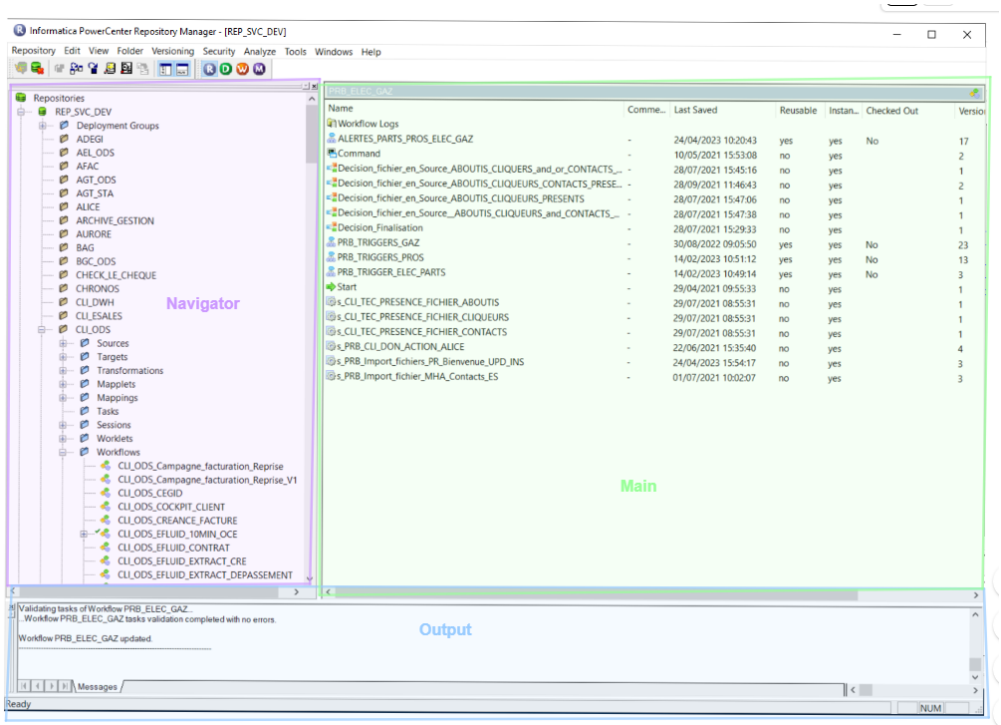


Figure 0.4.2: PowerCenter Repository Manager

The Repository Manager can display the following windows:

- Navigator. Displays all the objects you create in the Repository Manager, Designer and Workflow Manager. It is organized by repository and folder first.
- Main. Provides the properties of the object selected in the Navigator. The columns in this window change depending on the object selected in the Navigator.
- Output. Provides the output of the tasks performed in the repository manager.

PowerCenter Designer

The Designer allows you to create mappings by setting up transformation flows with sources and targets. There are several tools that allow us to do this process, here are some of the designer tools in their globality:

- Source Analyzer. Imports or creates source definitions.
- Target Designer. Imports or creates target definitions.
- Transformation Developer. Develops transformations to use in mappings.
- Mapplet Designer. Allows you to create sets of transformations for use in mappings.
- Mapping Designer. Create mappings that the integration service uses to extract, transform and load data.

The treatments can be realized in the workspace.

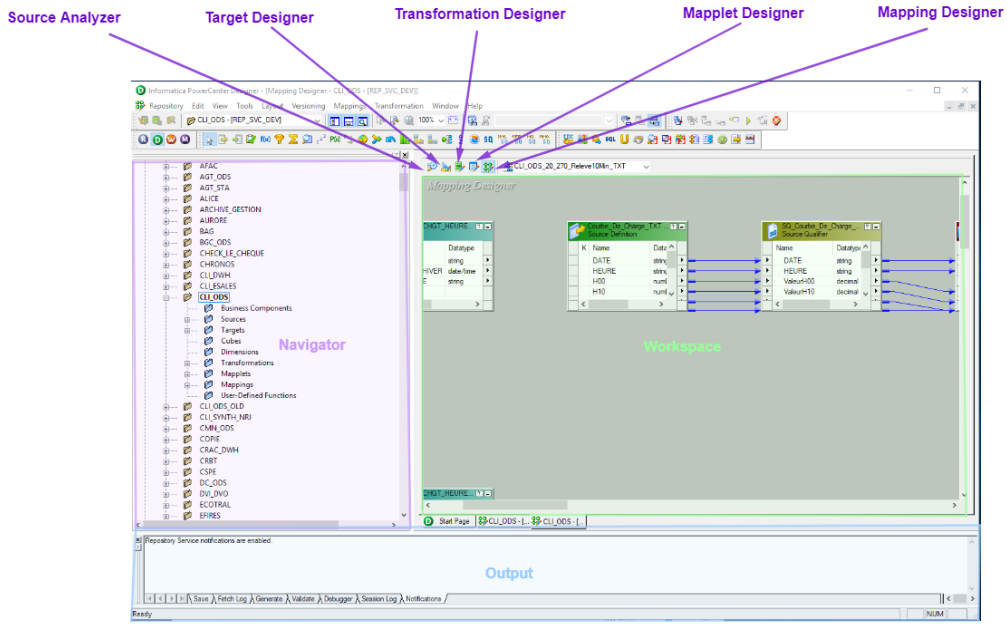


Figure 0.4.3: PowerCenter Designer

PowerCenter Workflow Manager

We have seen in the previous section that in general the designer is used to create mapping/mapplet. But it is possible to execute several instruction flows with the Workflow manager, in the workflow manager you define a set of instructions to execute tasks like sessions, emails and shell commands. The Workflow Manager has the following tools to help you develop a workflow:

- Task Developer. Create the tasks you want to accomplish in the workflow.
- Worklet Designer. A worklet is an object that groups together a set of tasks. A worklet is similar to a workflow, but without the scheduling information. You can nest worklets in a workflow.
- Workflow Designer. Create a workflow by linking tasks in the Workflow Designer. You can also create tasks in the Workflow Designer as you develop the workflow.

In order to create a workflow, the Workflow Manager has tasks like the Session task, Order task, and Email task. The Session task is based on a mapping that you build in the Designer. The tasks are then linked together using links to describe the order in which the tasks you created are carried out. Workflow branches can be made using conditional linkages and workflow variables. The integration service collects the necessary metadata from the repository prior to the workflow's start time in order to carry out its activities. The Workflow Monitor allows you to keep tabs on the workflow's progress.

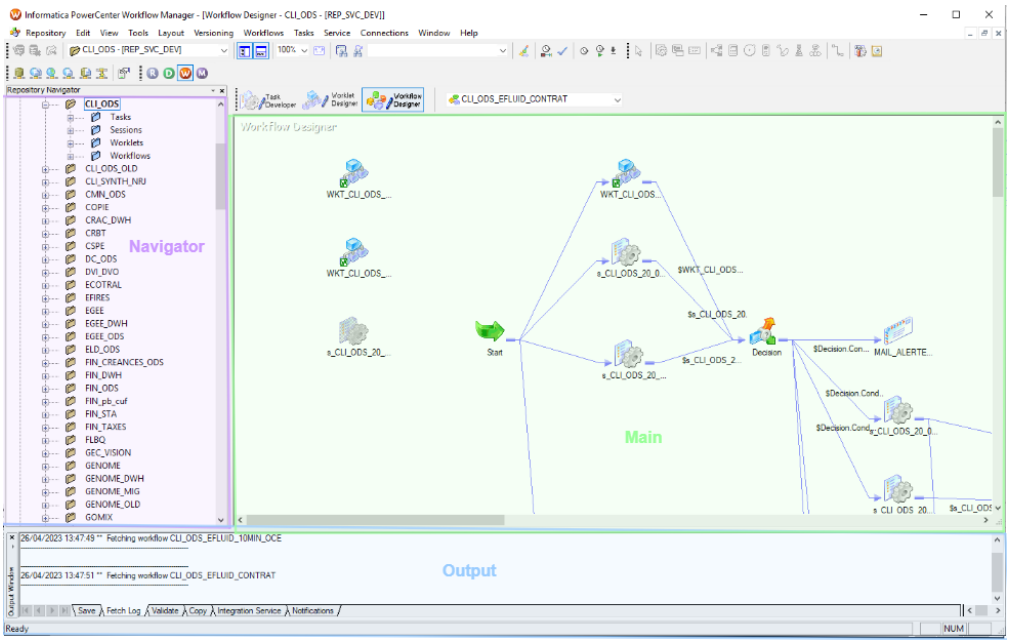


Figure 0.4.4: PowerCenter Workflow Manager

PowerCenter Workflow Monitor

After creating your workflows with the workflow manager, you can execute them with the workflow monitor.
In the Workflow Monitor, you can keep an eye on tasks and workflows. In the task view or Gantt Chart view, you can see information about a workflow or job. Workflows can be started, stopped, aborted, and resumed using the Workflow Monitor. It is also possible to consult the log of the executed workflows in order to see in detail all that occurred and the errors and warnings returned. The Workflow Monitor consists of the following windows:

- Navigator window. Displays monitored repositories, servers, and repositories objects.
- Output window. Displays messages from the Integration Service and Repository Service.
- Time window. Displays progress of workflow runs.
- Gantt Chart view. Displays details about workflow runs in chronological format.
- Task view. Displays details about workflow runs in a report format.

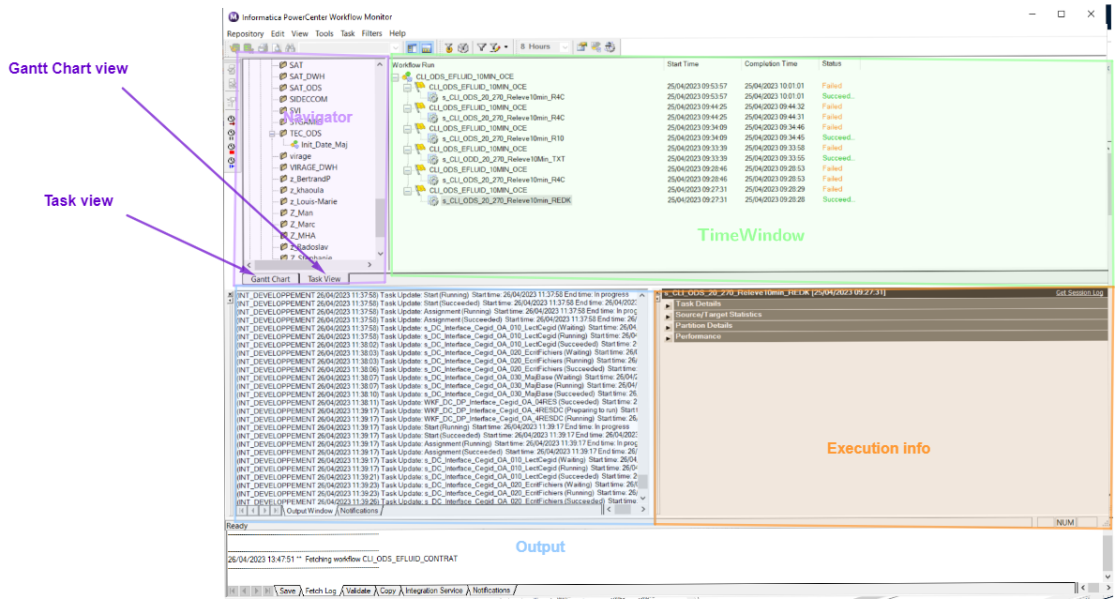


Figure 0.4.5: PowerCenter Workflow Monitor

0.4.3 Mapping

The heart of Informatica processes are mappings. Mappings are created in the Designer and then used in the workflow manager to associate them with sessions. Basically, a mapping is a set of sources to which transformations are applied in order to link them to the targets.

A mapping is made up of several essential elements. First of all, we have the source definition, which enables us to import sources from relationship tables, as well as from flat files, COBOL files or XML files.
Importing a source is done in the source analyzer. It's also possible to edit data types and their precision, as well as setting up primary and secondary keys if required.

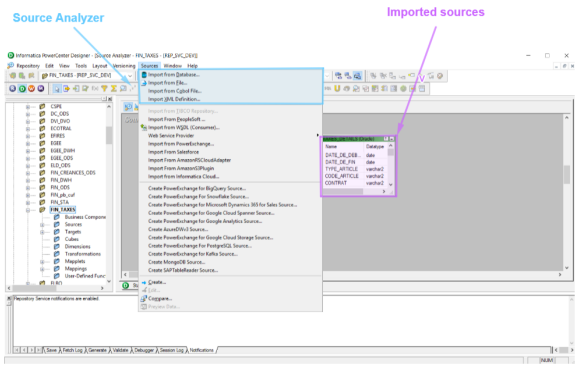


Figure 0.4.6: Import of sources

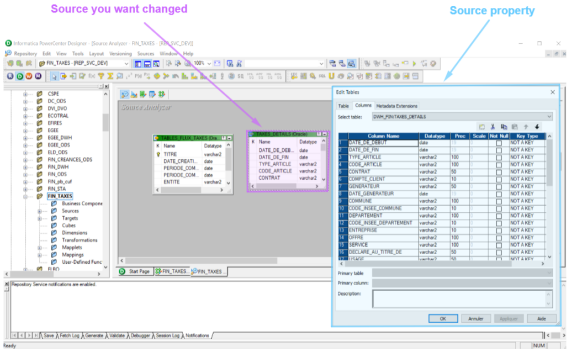


Figure 0.4.7: Edit Source

Another important component of a mapping is the targets, these are imported using the target designer and work in the same way as importing a source.

It's also possible to define parameters and variables when creating a mapping. As you can see from the photo, a parameter has already been set up in this particular mapping. Mapping Parameters and Variables are an optional user-defined data types. The Mapping parameters and variables are used to create temporary variable objects. It helps us to define and store temporary values during the processing of mapping data. This user-defined data type is designed for mapping.

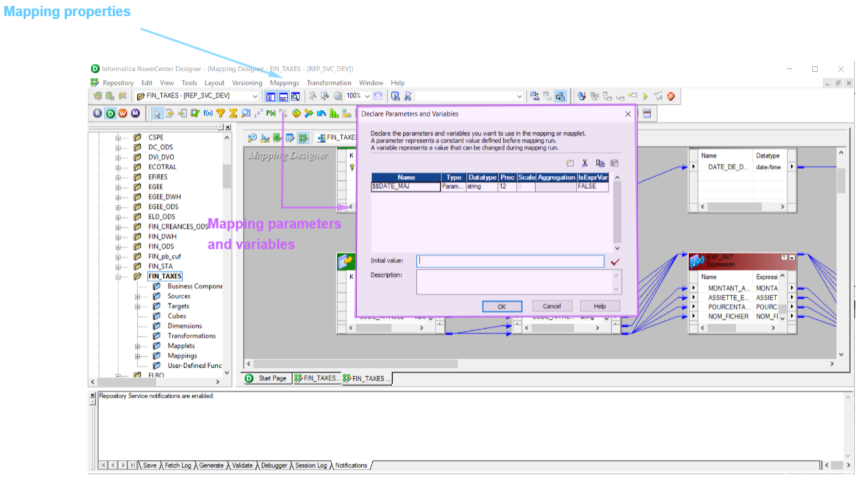


Figure 0.4.8: Parameters and variables of a mapping

We've seen that in order to create a mapping we need one or more sources and targets which will be linked together, but between these two elements it's possible to make changes and transformations to our data such as aggregations, joins or other elements defined by Informatica. Transformations are a part of a mapping that represent the operations that you want to perform on data. Transformations also define how data enters each transformation.

Transformation is classified into two categories-the first one based on connectivity.In Informatica, one transformation is connected to other transformations during mappings are called connected transformations.
Those transformations whose not link to any other transformations are called unconnected transformations.

The second category is based on the change in the number of rows.
First of all, we have the active transformations, these transformations change the amount of input data. For example, filter transformations are active transformations, the filter will remove data according to a specific condition. So we won't have the same number of input and output data.
We also have passive transformations, which keep the same number of input and output data, such as the expression transformations that we'll see further on.

Workflow

In Informatica, a workflow is a sequence of tasks or actions designed to carry out a specific data integration or transformation process. It is a major component of the Informatica PowerCenter tool, used for enterprise data integration and management.

There are several steps to creating a workflow, the first being to create the workflow in the workflow

manager, after that we can add several tasks to it. There are several types of tasks that can be used to carry out different actions.
Let's take a look at the most commonly used tasks when creating a workflow: :

1. First of all, we have session tasks, the type of task we use most in our treatments. This task executes individual sessions of a mapping. It controls and monitors the execution of a specific session, including the management of session parameters and the collection of execution statistics.

2. We also have decision tasks, which enable us to make a decision based on a specific condition. For example, it can check whether a source file exists before continuing the workflow in a certain direction.

3. Command tasks are also widely used, allowing system commands, scripts or external programs to be executed as part of the workflow. This can be useful for performing specific actions, such as creating a folder to store data files, or running a script to clean up temporary files after data processing.

4. We can also find e-mail tasks, which are more rare but very useful for important treatments. These tasks enable e-mail to be sent to specified addresses. For example, you can set up an e-mail task when an important process is interrupted during execution, so that the information reaches the person assigned directly.

0.5 Realizations

In this section we'll look at the different types of transformation most commonly used in mapping, and illustrate them with examples developed during the internship.

0.5.1 Fist request → Pilotage ESE : NRJ_OFFRE

Here's a description of the first request I'm going to present:
Add data fields:

- TAUX_DEROG_EV number(10)
- DELAI_VALIDITE number(10)
- DATE_FIN_VALIDITE date
- DATE_CALCUL date
- TYPE_FORMULE varchar(200)
- REGUL_SPOT varchar(200)

- VOLUME_TOTAL_PREVISIONNEL number(10)

Load with:

- TAUX_DEROG_EV = EC_ES_KND.V_SUIVI_OFFRES_TERMINEES_DET.TAUX_DEROG_ENERGIE_VERTE
- DELAI_VALIDITE = EC_ES_KND.V_SUIVI_OFFRES_TERMINEES_DET.DELAI_VALIDITE
- DATE_FIN_VALIDITE = EC_ES_KND.V_SUIVI_OFFRES_TERMINEES_DET.DATE_FIN_VALIDITE
- DATE_CALCUL = EC_ES_KND.V_SUIVI_OFFRES_TERMINEES_DET.DATE_CALCUL
- TYPE_FORMULE = E_ES_KND.V_SUIVI_OFFRES_TERMINEES_DET.TYPE_FORMULE
- REGUL_SPOT = EC_ES_KND.V_SUIVI_OFFRES_TERMINEES_DET.REGUL_SPOT
- VOLUME_TOTAL_PREVISIONNEL = EC_ES_KND.V_VOLUME_CONSO_ANNUEL_REPREV.VOLUME_TOTAL_PREVISIONNEL

The link between the tables is: references_esales.

The first step in this request is to add the new fields to the database, i.e. the Tableau ESE and the NRJ_OFFRE table. To do this, simply run this script on SQL developer:

```
ALTER TABLE nrj_offre ADD TAUX_DEROG_EV number(10);
ALTER TABLE nrj_offre ADD DELAI_VALIDITE number(10);
ALTER TABLE nrj_offre ADD DATE_FIN_VALIDITE date;
ALTER TABLE nrj_offre ADD DATE_CALCUL date;
ALTER TABLE nrj_offre ADD TYPE_FORMULE varchar(200);
ALTER TABLE nrj_offre ADD REGUL_SPOT varchar(200);
commit;
```

Next, we need to find out which mapping to modify. To do this, we will look at the folder where the mappings are located to see the dependencies between the targets and sources we have. In the request, we have information on the target to be modified, the NRJ_OFFRE table. Next, we'll see in which mapping this target is used.

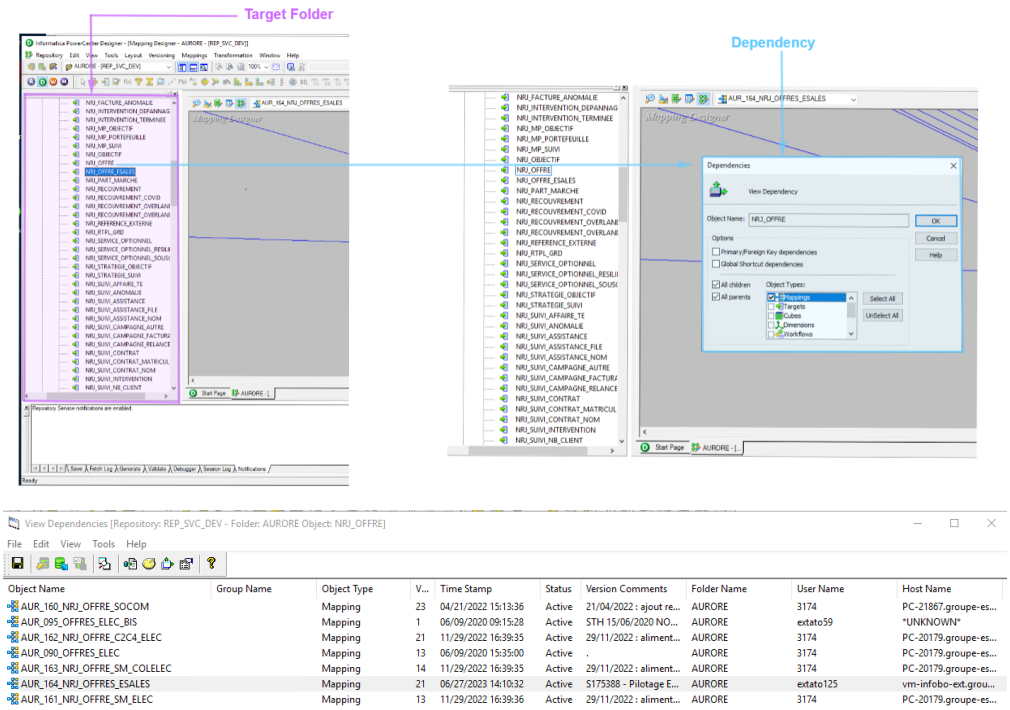


Figure 0.5.1: View target dependencies

As you can see, this target is used in several mappings, so the next step is to go through all these mappings in order to understand the processing performed and find out which one to modify.
Having found the mapping to modify, we'll first try to modify the processing by adding the two new sources to the source qualifier.
The source qualifier is one of the essential transformations when creating a mapping, and is often placed just after the source. It allows us to extract sources according to specific characteristics, such as joiners, filters or sql queries. The source qualifier allows us to bridge the gap between the source and other transformations.
Additionally, the source qualifier transformation plays a significant role in determining how the integration service reads data from the source system. It helps in identifying whether the data should be read row by row or using bulk mode, which can significantly impact performance and efficiency.
You may regulate the data taken from the source, perform initial filtering or cleansing, and specify the flow of data to future transformations in the mapping by setting out the source qualifier transformation. You can modify and prepare the data for further processing or integrating into a target system.
After the import of the two new sources,all we had to do was add the necessary columns to the

source qualifier, which had already been set up in the mapping.

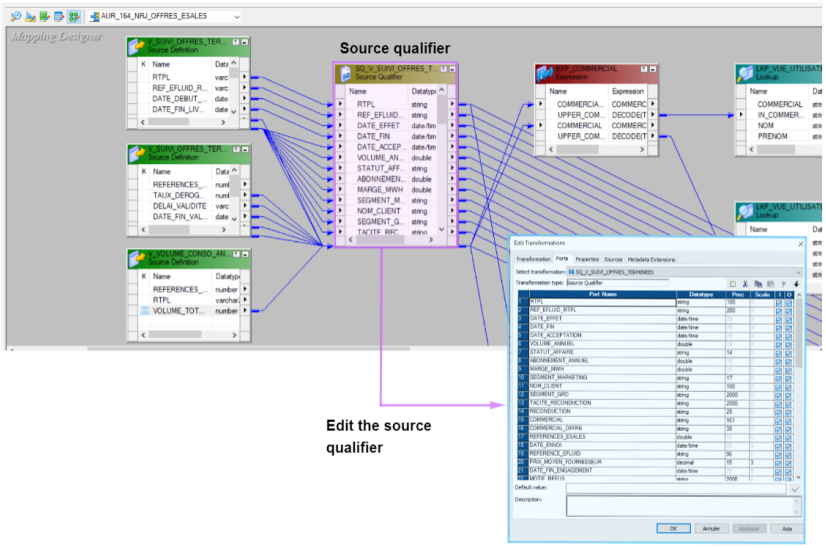


Figure 0.5.2: Parameters and variables of a mapping

The V_SUIVI_OFFRES_TERMINEES table was the only source for this mapping, due to the request I had to add the V_SUIVI_OFFRES_TERMINEES_DET and V_VOLUME_CONSO_ANNUEL_REPREV tables and feed the source qualifier with the new fields. You can see that the source qualifier is specified by a SQL QUERY. When you configure a source qualifier transformation, Informatica can generate the SQL query automatically based on the source definition. However, you also have the option to write custom SQL queries to extract the desired data from the source.
Beware when you make an override the mapping will only execute the query defined in this field, so if you want to add a joiner or a filter you must remember to put them in the SQL QUERY and not in the USER DEFIGNED JOIN or SOURCE FILTER.

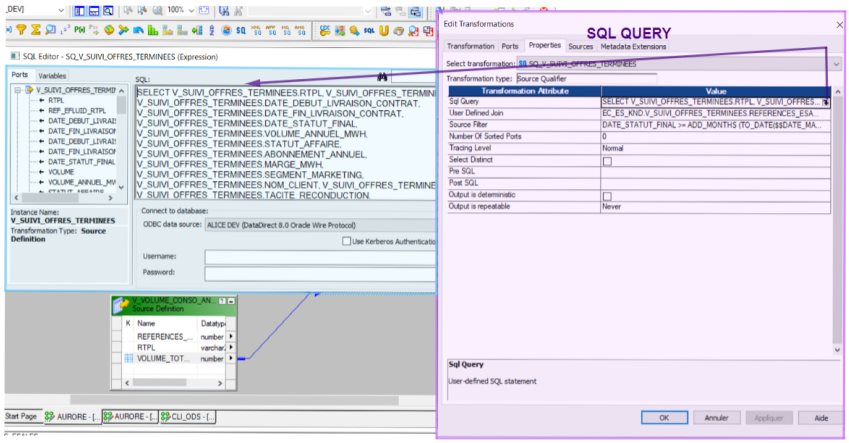


Figure 0.5.3: Parameters and variables of a mapping

Once we've made the necessary changes to the mapping, we can test our stategy by running the workflow where the session is located. After execution, we notice in the workflow monotor that this treatment takes a considerable amount of time and keeps running.
To understand the problem, the ideal solution would be to run the qualifier query directly in SQL developer in order to see if the problem is due to the modifications we've made. After testing that, we noticed that the problem came from the join on the new V_VOLUME_CONSO_ANNUEL_REPREV source, which was too slow, and we therefore decided to exclude this source for the moment, it will be enough to include only the V_SUIVI_OFFRES_TERMINEES_DET source and make a single join between V_SUIVI_OFFRES_TERMINEES_DET and V_SUIVI_OFFRES_TERMINEES.
To optimize the process, we also decided to treat the join in Informatica directly. To do this, we first sorted our two sources on the field references_esales, which we then used for the join to optimize the performance of our processing.

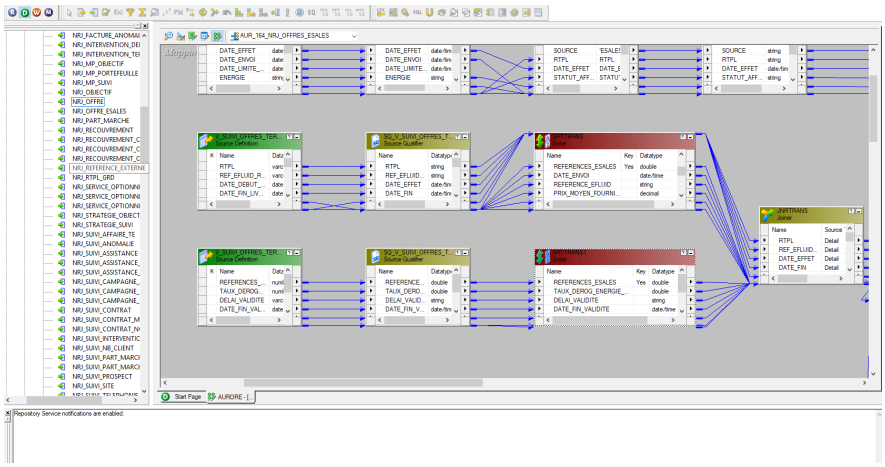


Figure 0.5.4: Update mapping with new treatments

0.5.2 Second request → FIN_TAXE : Creation of an aggregation

Here's a description of the second ticket I'm going to present:

Create a taxes_agg table with:
date_de_debut, date_de_fin, declare_au_titre_de, entreprise, commune, code_insee_commune, offre, prix_unitaire, assiette_a_declarer, montant_a_declarer, assiette_exoneration, pourcentage_exoneration, type_article, code_article, nom_fichier, usage, puissance_maximale, montant_tccfe, montant_tdcfe
Creation of an Informatica process with :

1. Delete data from the calculation period.

```
Select distinct periode_comptable_debut from tables_flux_taxes where
trunc(date_creation ,'DD') = trunc(sysdate,'DD')
```

2. Aggregation calculation:

```
select date_de_debut, date_de_fin, declare_au_titre_de, entreprise,
commune, code_insee_commune, offre, prix_unitaire,
SUM(assiette_a_declarer) assiette_a_declarer,
SUM(montant_a_declarer) montant_a_declarer,assiette_exoneration,
pourcentage_exoneration, type_article, code_article, nom_fichier,
usage, puissance_maximale,
sum(case when taxes_details.type_article = 'tccfe'
    then taxes_details.montant_a_declarer
    else 0
```

```
        end) montant_tccfe,
    SUM(CASE WHEN taxes_details.type_article = 'tdcfe'
        THEN taxes_details.montant_a_declarer
        ELSE 0
        END) montant_tdcfe
    FROM taxes_details
    WHERE date_de_debut in
    (SELECT distinct periode_comptable_debut FROM tables_flux_taxes
    WHERE trunc(date_creation ,'DD') = trunc(sysdate,'DD'))
    GROUP BY date_de_debut, date_de_fin, declare_au_titre_de, entreprise,
    offre,commune, code_insee_commune, prix_unitaire, assiette_exoneration,
    pourcentage_exoneration, type_article,code_article, nom_fichier,usage,
    puissance_maximale;
```

The first step is to create the new table. As you can see, there are no precise data types, which is why you need to look at the feed of each field to find the type and precision of the data in the new table.

```
CREATE TABLE TAXES_AGG
(
    date_de_debut date,
    date_de_fin date,
    declare_au_titre_de varchar2(50),
    entreprise varchar2(10),
    commune varchar2(100),
    code_insee_commune varchar2(10),
    offre varchar2(100),
    prix_unitaire number(9,5),
    assiette_a_declarer number(15,2),
    montant_a_declarer number(15,2),
    assiette_exoneration number(12,2),
    pourcentage_exoneration number(5,2),
    type_article varchar2(100),
    code_article varchar2(100),
    nom_fichier varchar2(100),
    usage varchar2(50),
    puissance_maximale varchar2(50),
    montant_tccfe number(15,2),
    montant_tdcfe number(15,2)
);
commit;
```

Once the table has been created, we can start processing it in Informatica. To do this, we'll use the designer to create a new mapping.

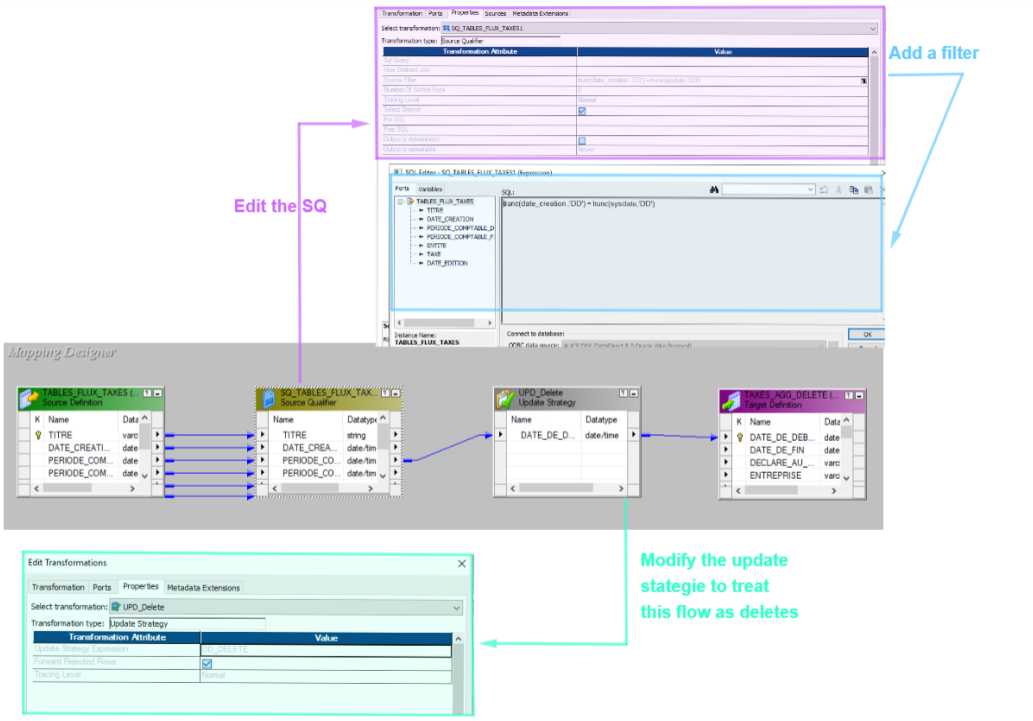


Figure 0.5.5: Treatment flow for executing deletes

First of all, we're going to start by processing the deletes. This flow is necessary to avoid duplicates in our table, as we're going to create an aggregation that SUMs values over a well-defined period of time, we need to be careful not to recalculate this aggregation for the same period of time. The third transformation in this flow corresponds to an update strategy. Update Strategy is a transformation in Informatica that controls how rows of data are processed when data is updated or inserted into a target. There are several ways of processing data in the target, starting with the "DD_INSERT" insert strategy (to insert a new row), the "DD_UPDATE" update strategy (to update an existing row), the "DD_DELETE" delete strategy (to delete a row) and the "DD_REJECT" reject strategy (to reject a row).

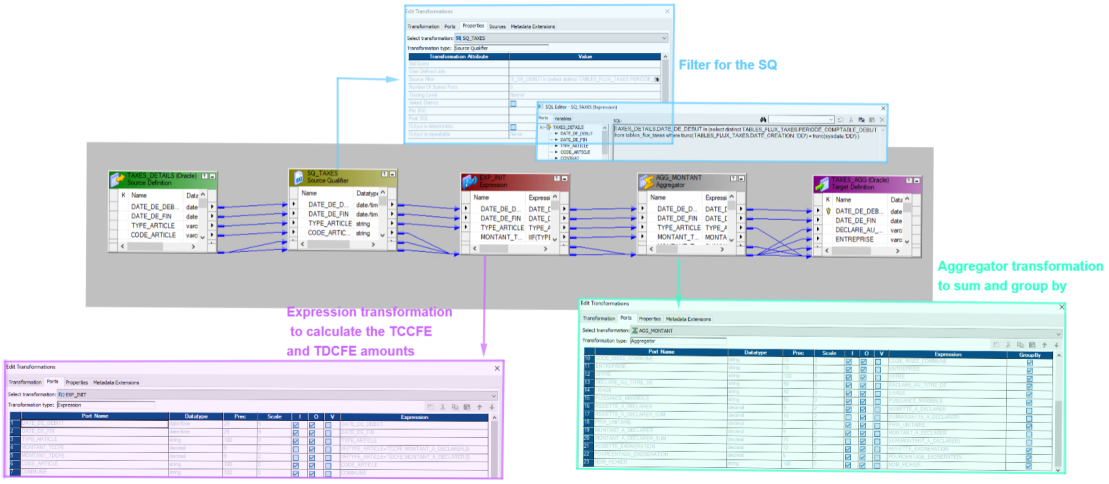


Figure 0.5.6: Treatment flow for the aggregation

After we've made the deletes, we can proceed to the flow to perform the aggregations. First of all, as in all mapping, we start with the SQ, in which we add a filter that specifies the period of time on which we want to perform the aggregation. We then use an expression transformation to create two new fields, montant_TCCFE and montant_TDCFE, based on the MONTANT_A_DECLARER field. We then need to perform the aggregation, for which we have a transformation that enable us to perform this type of transformations. Thanks to this transformation, we'll perform the summations on montant_TCCFE and montant_TDCFE and also specify which fields we're grouping by. The last transformation just before the target is a update stategy .

When you configure the session, consider the key constraints along with the duplicate row handling option for an effective update strategy. For the target table, you can choose between insert, update as update, update else insert, update as insert, or delete. The update else insert option has a very poor performance. Instead of using the update else insert option, Informatica advises using the update as insert option.

The PowerCenter Integration Service stamps each row with an indicator when reading source data to indicate which operation should be carried out when the row reaches the target. You can set this value on the Task tab by selecting one of the Treat source rows as options, such as insert, update, delete, or data driven.

The following table describes the values of Treat Source Rows As Options:

Treat Source Row As Options	Description	Recommendation
Insert	Marks all rows to insert into the target.	Turn on the Insert flag in the target table property.
Delete	Marks all rows to delete from the target.	Turn on the Delete flag in the target table property.
Update	Marks all rows to update into the target. You can further define the update type in the target.	Turn off the Insert and Delete flags in the target, and select any type of update in the target table.
Data Driven	The PowerCenter Integration Service uses Update Strategy transformations in the mapping to determine the operation on a row-by-row basis. You can define the update operation in the target options.	If the mapping contains an Update Strategy transformation, the default option is Data Driven. You can also use this option when the mapping contains Custom transformations configured to set the update strategy.

Figure 0.5.7: Treat Source Row As Options

In the case of this request we preferred to choose the data driven mode.

0.5.3 Third request → Pilotage ESE : Creation nrj_suivi_volume_projete_hdp

Here's a description of the third ticket I'm going to present:

- Table:
  1. Add a nrj_suivi_volume_projete_hdp table identical to the volume_projete table.
  2. Add the date_suivi field to nrj_suivi_volume_projete_hdp
- Traitement:
  1. Delete the nrj_suivi_volume_projete_hdp lines where date_suivi = last_day($$DATE_MAJ)
  2. Insert in nrj_suivi_volume_projete_hdp the lines volume_projete where segment_marketing not in ('PARTICULIER', 'PROFESSIONNEL')
  3. Add date_suivi = last_day($$DATE_MAJ)
  4. Add session in WKF_PILOTAGE_MP after session s_AUR_330_PERTES_PREVISIONNELLES_GAZ

The first step consists in creating the table in the database.

```
CREATE TABLE NRJ_SUIVI_VOLUME_PROJETE_HDP
(
      DATE_SUIVI DATE,
      TYPE VARCHAR2(50),
      RTPL VARCHAR2(50),
      DATE_DEBUT DATE ,
      DATE_FIN DATE,
      DATE_FIN_CALCULEE DATE ,
      VOLUME NUMBER(15,2) ,
      SEGMENT_MARKETING VARCHAR2(50) ,
      ENERGIE VARCHAR2(50) ,
      REFERENCE_CONTRAT VARCHAR2(50) ,
      STATUT_CONTRAT VARCHAR2(50) ,
      PUISSANCE_SOUSCRITE NUMBER(15,2) ,
      SERVICE VARCHAR2(50) ,
      DATE_CESSATION DATE ,
      OFFRE_PRODUIT VARCHAR2(50) ,
      SEGMENT_GRD VARCHAR2(50) ,
      MARGE_ABONNEMENT NUMBER(15,2) ,
      MARGE_MWH NUMBER(15,2) ,
      MARGE_ABO_CALCULEE NUMBER(15,2) ,
      MARGE_MWH_CALCULEE NUMBER(15,2) ,
      DATE_FIN_LIVRAISON DATE ,
      NB_CONTRATS NUMBER(15,2) ,
      VOLUME_CALCULE_N NUMBER(15,2) ,
      VOLUME_CALCULE_N_1 NUMBER(15,2) ,
      VOLUME_CALCULE_N_2 NUMBER(15,2) ,
      VOLUME_CALCULE_N_3 NUMBER(15,2) ,
      MARGE_CALCULEE_N NUMBER(15,2) ,
      MARGE_CALCULEE_N_1 NUMBER(15,2) ,
      MARGE_CALCULEE_N_2 NUMBER(15,2) ,
      MARGE_CALCULE_N_3 NUMBER(15,2) ,
      PROFIL VARCHAR2(10) ,
      VOLUME_CALCULE_N_4 NUMBER(15,2) ,
      MARGE_CALCULEE_N_4 NUMBER(15,2) ,
      DUREE_AN VARCHAR2(10) ,
      SEGMENT_SPECIFIQUE VARCHAR2(50) ,
      DATE_ACCEPTATION DATE
) ;
commit;
```

In this process, we use a variable called date_maj, which allows us to run processes according to an update date, or to track dates for certain processes.
In the case of this request, the date_maj is used more as a tracing date, which allows us to have a tracing by month for this mapping, but it's also possible to have a tracing by year or by week, depending on the request.
Let's try to understand the principle of date_suivi based on this request. As in this case we want to track by month, we first need to delete all data for which the date_suivi corresponds to the month of the current date. We then run the process again for the same date_maj , and finally change the date_maj to the current date.
Here's an example explaining the process

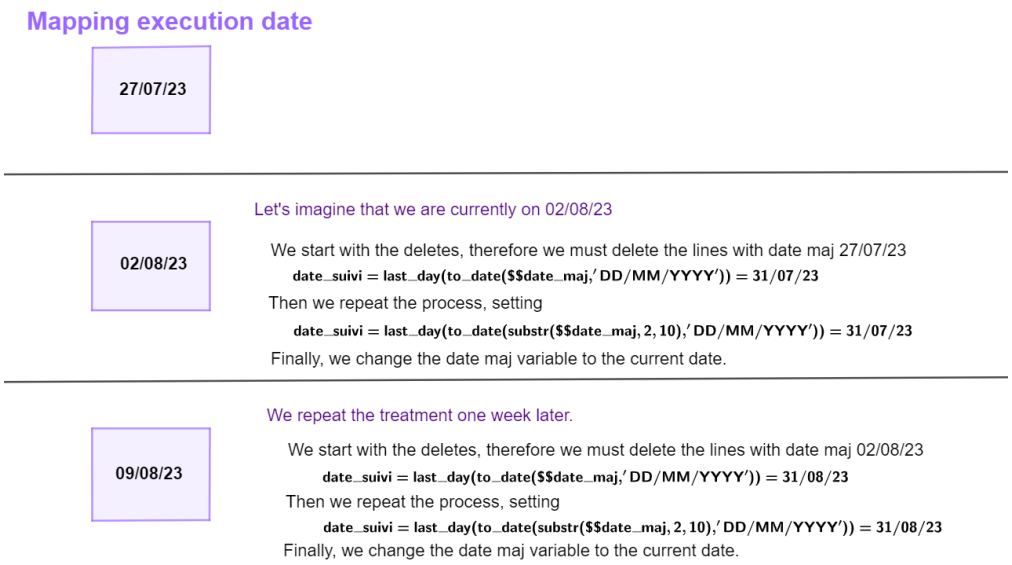


Figure 0.5.8: The Update Date Process

Now let's take a look at the mapping implemented in Informatica.
In this first data flow we delete the data lines that have been modified this month based on the last tracking date added.

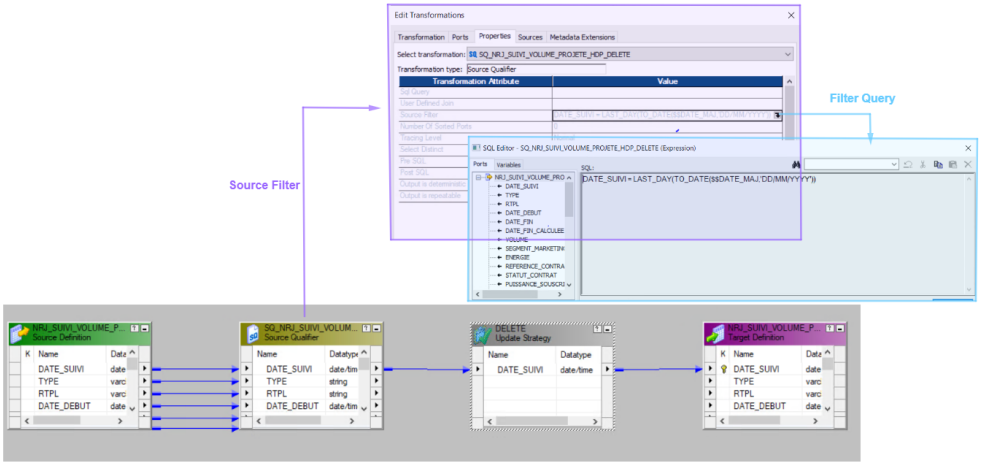


Figure 0.5.9: Delete Process

We then add the data by changing the tracking date to the last day of the month of the last update date.

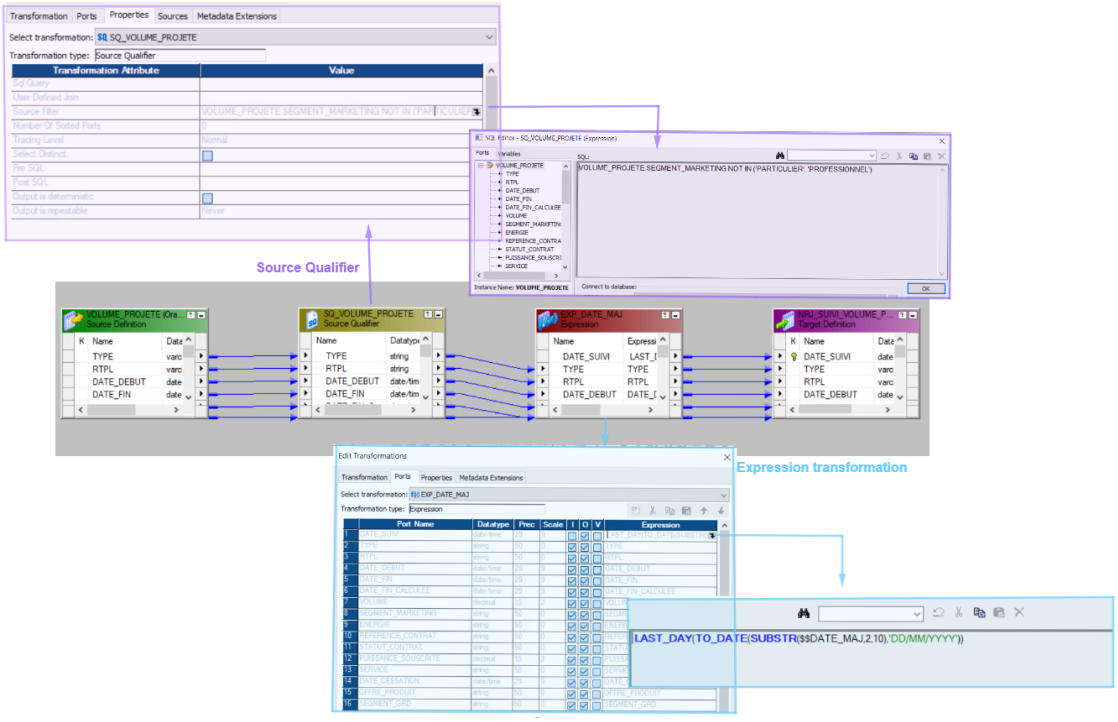


Figure 0.5.10: Insert Process

After the deletes and inserts, the next step is to change the update date in the database so that it is taken into account for future sessions. To do this, we use a mapplet, which takes into account a 'session_name' variable corresponding to the name of our session. The mapplet searches the database for fields with the name of the executed session, then changes the update date to the current date. This table is therefore modified each time the session is executed.

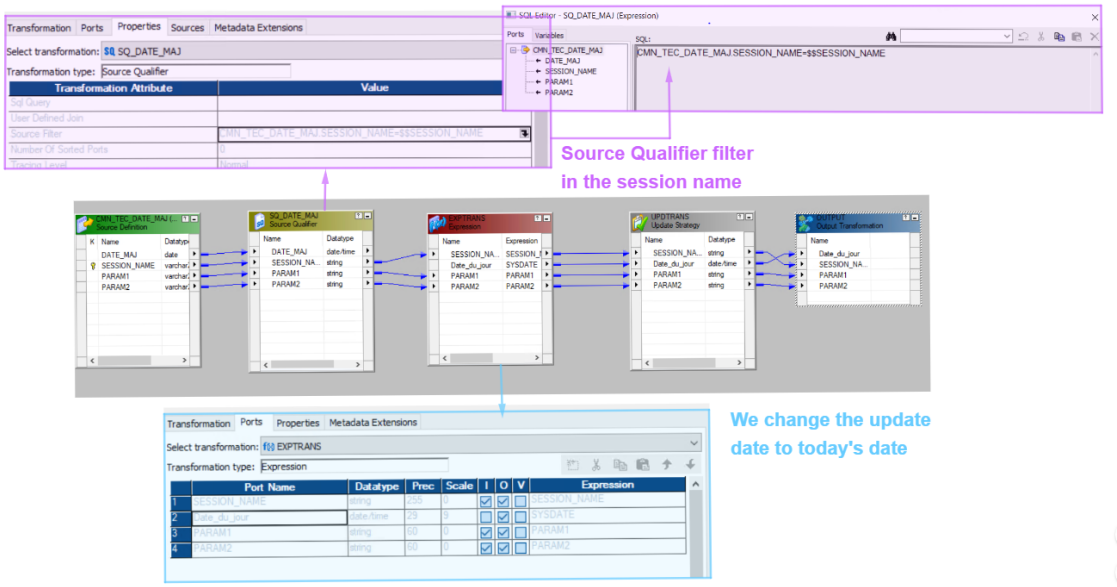


Figure 0.5.11: MAJ Mapplet

Thanks to this table, we can check session execution dates, or view maj dates.

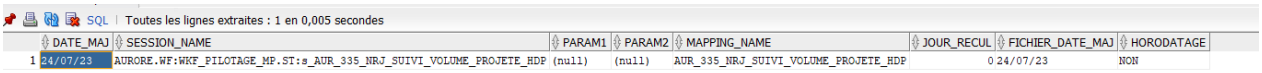


Figure 0.5.12: Database ETL_TEC

IMMO Project

During this internship I also had the opportunity to work on the IMMO project, which was already in place and being handled by one of my BI colleagues. I had to process a series of 3 tickets.
The first ticket and the most interesting one of this group corresponded to an incidence: there was a volumetric problem when writing data to Oracle. The Cegid XRP immo files generated entries for endowments, commissioning and withdrawals. Commissioning and withdrawal entries are individualized. The problem was that Cegid XPR generated files with more than 50,000 lines, whereas oracle would only accept 20,000 lines at a time. In this case, it would be ideal to create

an automatic splitting mapping of the 1x global entries file, into slices of up to 20,000 after sorting by on Journal No. (col 1 to 2), then on entry number (operation label field Pos start 129, pos end 178, length 50), breakage criterion and therefore counterparty.
The splitting process must be a new, independent process to be run after the existing process, so that it can be reused when integrating other high-volume files. Here are the constraints to be taken into account when splitting.

1. Before processing, sort 4RESDC records by journal code (col 1 to 2) and by entry number (Operation label field, position 129 to 178, length 50).
2. Rules for writing files :
   - There must be no more than 20,000 records per file (usually 48 is the largest).
   - If a file for a journal code contains more than 20000, it should be split into another file, but never separate the records of an accounting entry: records with the same operation label.

To summarize the request, we had to create a process that would split the data into several parts. To do this, we had to be careful not to have more than 20,000 pieces of data per file, but also to ensure that the data with the same label operation remained in the same file,
The first step is to split the workflow that processes the 3 companies into 3 different workflows so that there is only one workflow for each company. The 3 workflows are quite similar, so I'll only present the RESD case.

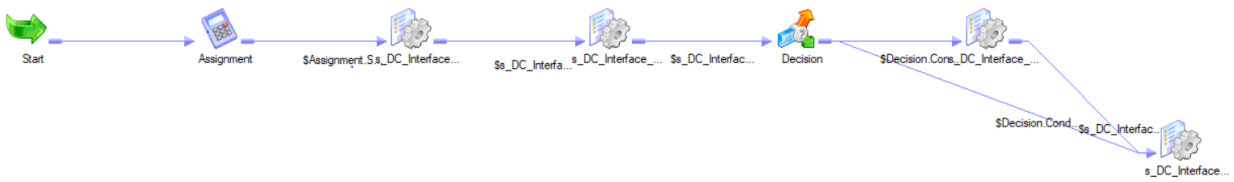
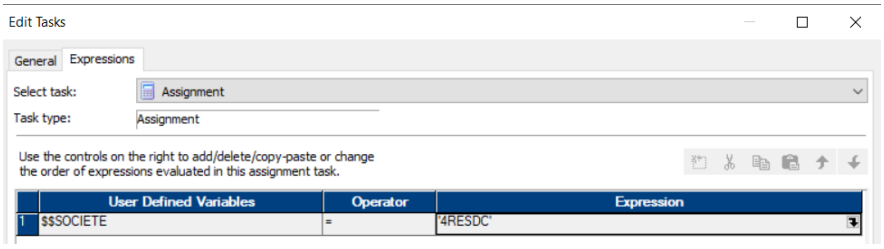


Figure 0.5.13: RESD Workflow



We can observe that the workflow begins with an assignment task. This will allow us to create a variable that can be used in the mapping associated with the workflow session.

Figure 0.5.14: Assignement Task

The first mapping "DC_Interface_Cegid_OA_010_LectCegid" consists in retrieving information from the database by filtering on the company name variable. Then we execute the mapping DC_Interface_Cegid_OA_020_EcritFichier, which takes the data of interest from the file created in the mapping "DC_Interface_Cegid_OA_010_LectCegid".

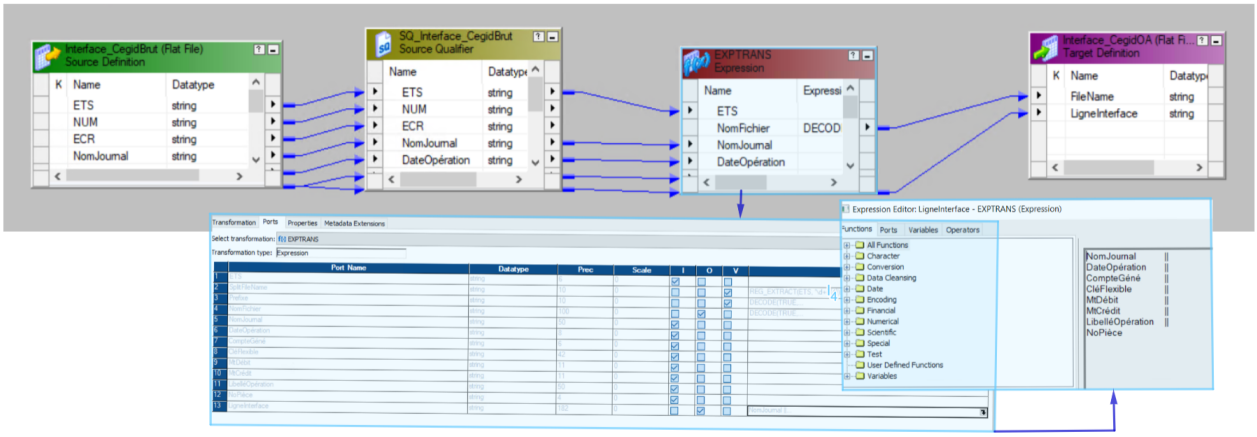


Figure 0.5.15: Mapping DC_Interface_Cegid_OA_020_EcritFichier

In this mapping, we can see that there are only 2 columns, Filename and LigneInterface. LigneInterfaces corresponds to a concatenation of all the data that interest us, this concatenation is made in the previous transformation. The filename corresponds to the name given to the generated file. In order to generate the filename dynamically, a new type of column, "FileName column", must be added to the target when the target is created. This creation will have no impact on target properties in the session.
We can see in the session properties that there is another file generated, but this one will be empty because the data will already be stored in the file generated at runtime.

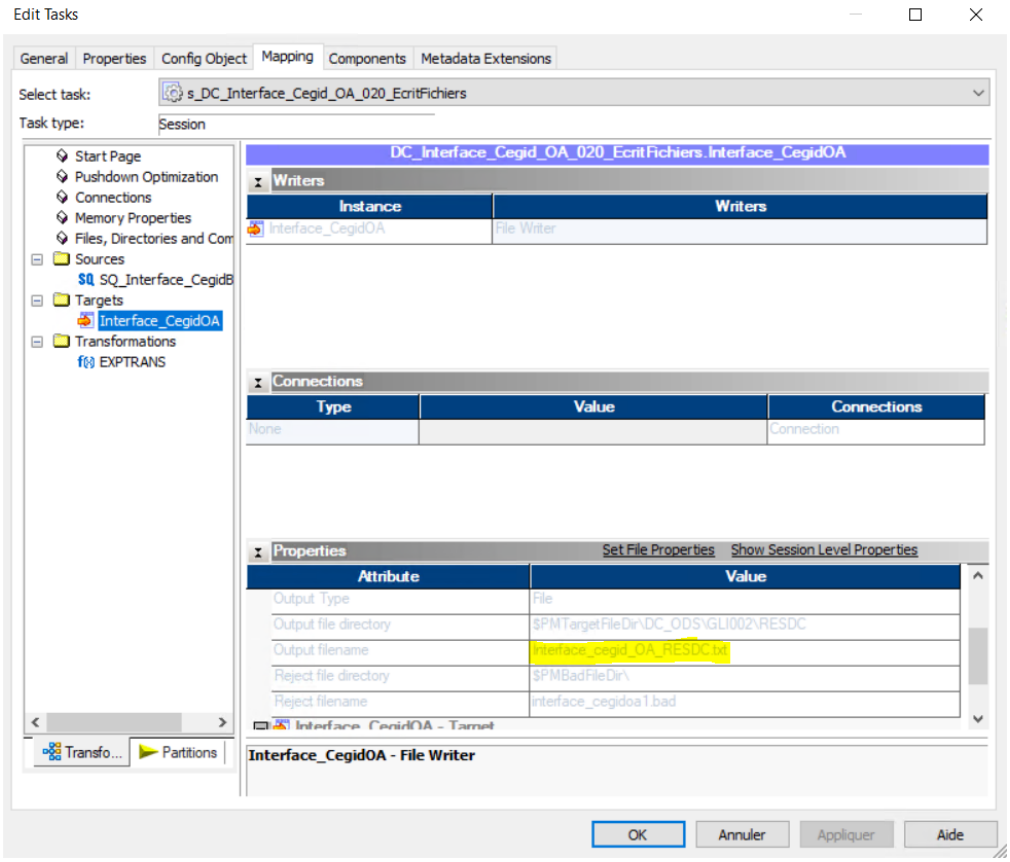


Figure 0.5.16: Session properties of the target

Finally we have splitting, which takes the RESD file as its source and splits it.

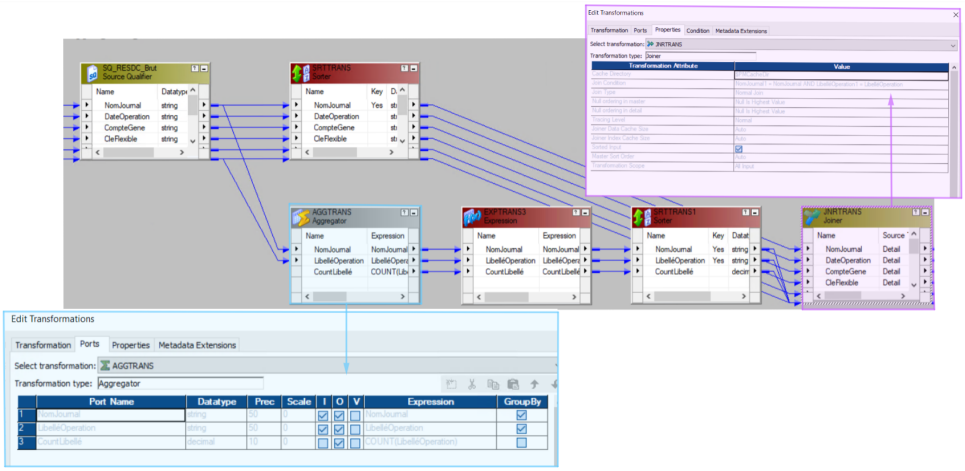


Figure 0.5.17: Fist part of the splitting mapping

First of all, we perform a sorter based on the NomJournal and LibelléOperation fields. At the same time, we perform an aggregation count(LibelléOpetation) on the libellé by doing a group by on NomJournal and LibelléOperation. We therefore calculate the number of LibelléOperation by NomJournal and LibelléOperation and then perform an sorter on the same two fields. The two sorts done in the two feeds in parallel will allow us to perform a join on the fields LibelléOperation and NomJournal in order to have the count(LibelléOperation) on each field.

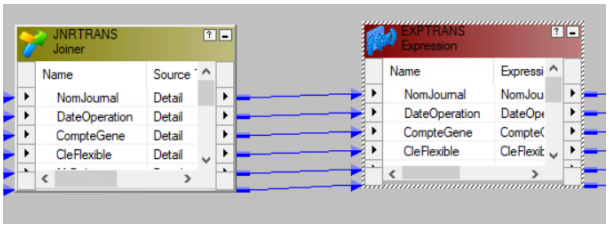


Figure 0.5.18: The first expression transformation

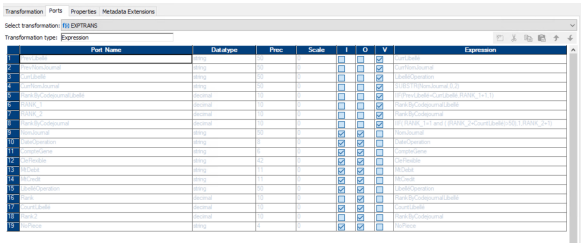


Figure 0.5.19: Properties of the EXTRANS

Then we have a transformation expression, which will first of all perform a computation on the libellé. If the libellé of the previous line is equal to the libellé of the current line, we'll add +1 to our Rank, otherwise we'll go back to 1. Then we'll store the computed number in the Rank_1 field to reuse it in future computations. The order in which we place the new calculated fields is crucial, as it's used to make comparisons between the lines (current data/previous data) in order to create "counters". The processing performed is therefore used to count the libellé field; when the label changes, this counter is reset to zero.
Then we have the RankbyCodeJournal field (also called Rank_2), which is used to count lines in general. We count the lines of data inserted one by one and when the libellé is equal to 1 and when

(Rank_2+CountLibellé)>50 '(where Rank_2= RankbyCodeJournal) then we reset the counter to 0, otherwise we add 1 to it. This calculation allows us to determine on which line to perform the spitting, we look to see if Rank_1 is equal to 1 in order to have fields with the same libellé on the same file. We also check the case where we would have integrated the next batch of fields with the same libellé, i.e. we look at the case where we would have taken the next batch of libellé in order to test if we exceed the number of lines fixed per file (in the general case it's 20000 but for the tests we use 50).

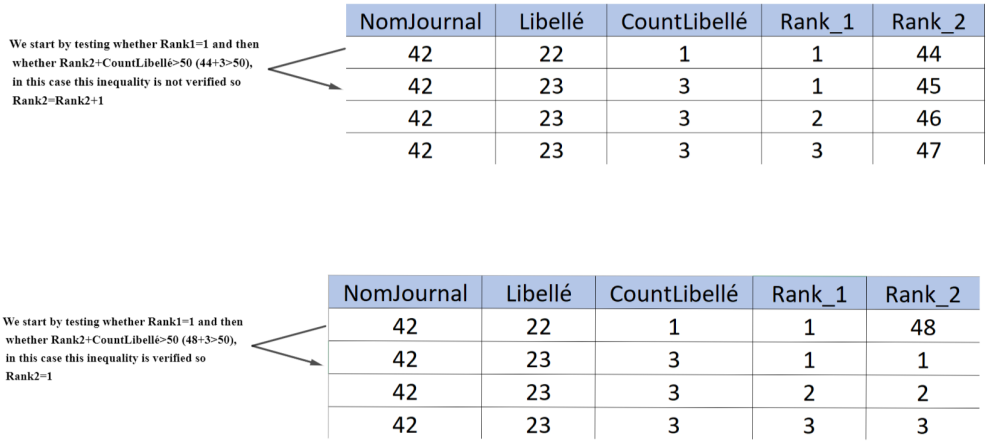


Figure 0.5.20: Two different scenarios

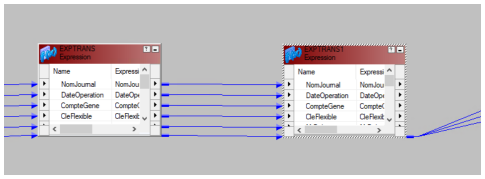


Figure 0.5.21: The second expression transformation

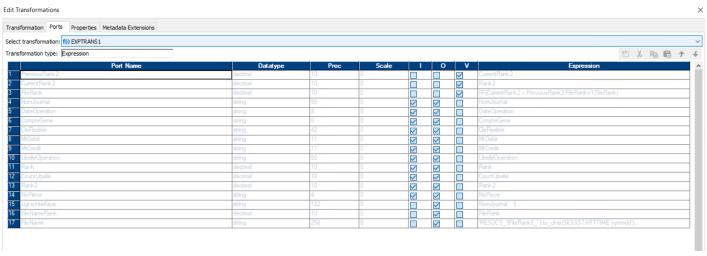


Figure 0.5.22: Properties of the EXTRANS1

In the following expression, we'll need to specify the FileName. We can see that we've created a new FileNameRank field which will allow us to create a counter for naming our splitting files.

NomJournal	Libellé	CountLibellé	Rank_2	FileNameRank
42	21	1	47	1
42	22	1	48	1
42	23	3	1	2
42	23	3	2	2
42	23	3	3	2

Figure 0.5.23: An exemple scenario

Notice that all the data of the first split file have a FileName of 1, all the data of the second split file have a FileName of 2. This column is then used in the FileName which corresponds to the name of the files created during the split.
The last transformation will allow us to perform splitting with the condition below:
If(ToChar(Rank2)='1',TC_COMMIT_BEFORE,TC_CONTINUE_TRANSACTION)

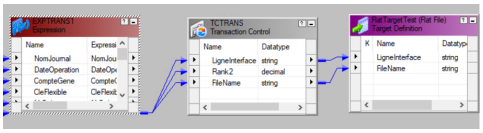


Figure 0.5.24: Transaction Control Transformation

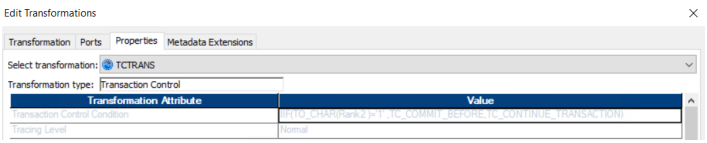


Figure 0.5.25: Splitting Condition

0.6 Conclussion

In conclusion of this Business Intelligence internship with Atos, it is clear that this experience has been extremely enriching and formative. During this period, I had the opportunity to learn to use new software, to understand and analyze customer needs and to progress in the field of BI.

It wasn't an easy start, with lots of new concepts to integrate, a well-defined procedure and a well-organized way of working, but thanks to my colleagues in BI, I was able to overcome all these challenges and gain access to all these new skills.

Working with ETL tools such as Informatica gave me an in-depth understanding of processes and data flows. I learned to identify and solve potential problems related to data quality and integrity, adapting my technical skills to the specific needs of the customer. I also learned to navigate Oracle databases, design robust database schemas and implement complex SQL queries to extract relevant information. During my time with Atos, I was gradually able to master the complex and essential aspects of data manipulation and optimization within an Oracle environment.

I would like to express my gratitude to the team for their guidance, shared expertise and constant support. Their guidance has greatly contributed to my professional development and my in-depth understanding of Business Intelligence processes.

Finally, I'm happy to announce that my involvement and results during this internship have been recognized, and I've been offered a permanent contract within the company. This opportunity will enable me to continue developing in the field of Business Intelligence, deepening my skills and actively contributing to the company's future projects.

Bibliography

[1] Es. https://fr.wikipedia.org/wiki/%C3%89lectricit%C3%A9_de_Strasbourg.

[2] Etl. https://www.oracle.com/fr/database/processus-etl-definition/.

[3] Informatica documentation. https://docs.informatica.com/.

[4] Rapport de stage de leslie.

[5] Sap documentation. https://help.sap.com/docs/SAP_BUSINESSOBJECTS_BUSINESS_INTELLIGENCE_PLATFORM?locale=en-US.

[6] Sql developer documentation. https://docs.oracle.com/cd/E12151_01/index.htm.

[7] Website of atos. https://atos.net/fr/.