Data warehouse with ETL
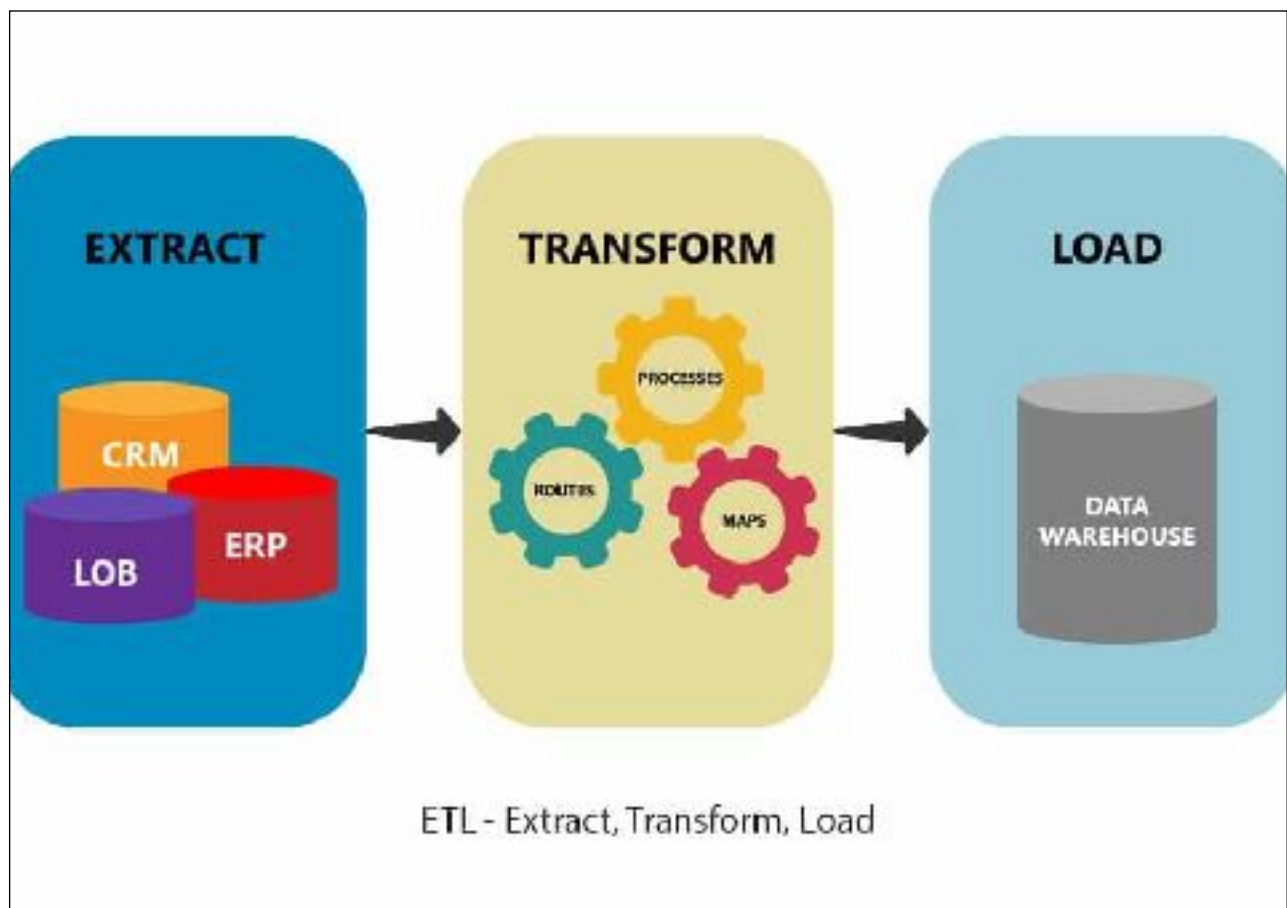
PROJECT

Sunday - 19 juin 2022

TEAM

- ❖ Enoch Foli-Degbesse
- ❖ Stephane Ganhi
- ❖ Thiam Hamidou
- ❖ BA Mouhamed
- ❖ Soheil Namdar Fard

Introduction

Data warehouse (DW) is pivotal and central to BI applications in that it integrates several diverse data sources, mainly structured transactional databases. The BI application then uses the transformed and formatted data to make power and insightful discovery from the data to aid top managers to make quality decision. However, the IT company ServiceSpot receives a lot of calls from their customers which are organized in several file formats CSv, Excel etc and would like to make good use of them in order to study and analyze them. Our goal is to implement an ETL process with SSIS that will retrieve data from its original format and load it into a new data warehouse. To do this, 3 steps are necessary :

- ❖ Data staging
- ❖ Operational Data store (ODS)
- ❖ Data warehouse

1 Objective

The goal of this project is to develop a Data warehouse system for the business intelligence team. This allows the companies to develop a system to support the decision makers and business strategist in making better decision using operational data.

2. Design of the ETL Process

The company has different Online Transaction Processing (OLTP) databases to extract, thus exist in excel and csv format. The ETL is the process of retrieving and transforming data from the source system and putting it into the data warehouse. With the scope of this project, we used SQL Server Integration services (SSIS) to design an ETL process to load the data into the data store (MSSQL Server)

2. 1 Staging

In the first step of this project which is staging, we use Excel and csv data base which are Years, Employees and Geography.

We get these data and put it in Stage package. Then arrange data work flow with sql queries. We named data as STA which is acronym for staging. We have four years 2018, 2019, 2020 and 2021 as ours transactional data. '**Ff**' stands for '**Flat file**' and '**dcnv**' 'data conversion'. Next we have to uniformise the intermediate tables.

It was necessary to convert the data recorded in English or French format and enter them into a table 'join all data' in order to consolidate data from Excel files. By click on 'All Data Join' we can see it. Thus we find the departure destination on the left and the arrival destination on the right. At the end of this step we retrieve the source data and put it in staging. It's evident that we do the same operation with Employees and Geography.

2. 2 Operational Data Store (ODS)

The operational data store (ODS) component is an architectural component of a data warehouse that is used for immediate reporting with current operational data. An ODS contains lightly transformed and lightly integrated operational data with a short time window. It is used for real time and near real time link between the Staging area and the data warehouse. Per our architectural design the ODS component had two tables: ODS_Employees and ODS_CallData. The following steps were considered to achieve the design :

ODS_Employee
- ❖ Data sourced from the staging area: STA_Employees
- ❖ The site field in the Employee table had both the state code and the State name has one data. In order to merge the employee and USStates table we needed to split the site field into states and state-code.
- ❖ Transformed the employeeID into Integer, and checked for any data that does not meet the format.
- ❖ Resizing the columns length to optimize the performance of the database.
- ❖ Final data stored in the ODS database, and the table called Employee

ODS_CallData
- ❖ Date sourced from the staging area: STA_CallData
- ❖ DateTime field in CallData table transformed into two separate fields; Date and Time
- ❖ Generation an SLA field bas on the requirement of waiting time with 35 secs is termed as "within SLA" otherwise termed as "Outside SLA »
- ❖ Merge the "data" and "callcharges" to become one table using the field "call-type" as the link.
- ❖ Checking for possible data conversion errors, if any is stored in the functional reject table for further analysis by the business/process owners
- ❖ Resizing the columns length to optimize the performance of the database.
- ❖ Final data stored in the ODS database, and the table called CallData.

2. 3 Dimension and fact table

In this third and last part of the project, the idea is to create a data warehouse in which we will store our information such as sources and targets in the form of a table. Its objective is to aggregate and enhance the data from these different sources. For its creation, we started by creating a new user where it was a question of defining the name of the data warehouse as well as the associated password for security and finally managing the access right. Once that was done, all we had to do was configure our connection to our data warehouse. This allows us to subsequently have a data warehouse organized by theme, namely DimDates, FactCallData and DimEmployees which have been formatted and finally unified to ensure consistency while maintaining non-revocation, ie maintaining the traceability of information and decisions taken. Indeed, the data is neither modified nor deleted. Thus a query issued on the same data several months apart will give the same results.
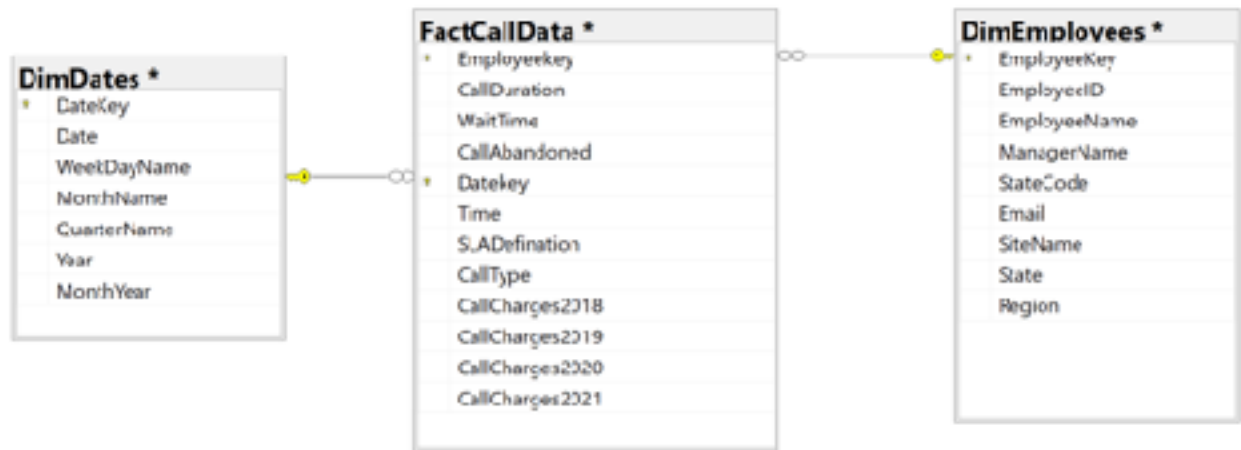
FIGURE A: Physical Design of the DWH